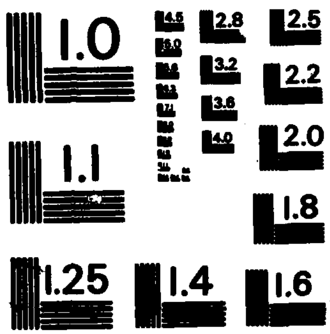
AD-A121 981 RIME: THE RECOVERABLE ITEM MANAGEMENT EVALUATOR VOLUME 1/2
I MODEL DESCRIPTION(U) DECISION SYSTEMS DAYTON OH
W S DEMMY MAY 80 TR-80-01 F33600-78-C-0524

UNCLASSIFIED F/G 9/2 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5

PHOTOGRAPH THIS SHEET

AD A121981

DTIC ACCESSION NUMBER

II

LEVEL

INVENTORY

RIME: Vol. I

DOCUMENT IDENTIFICATION

Rept. No. TR-80-01

Contract F33600-78-C-0524 May 80

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

ACCESSION FOR	
NTIS GRA&I	☒
DTIC TAB	☐
UNANNOUNCED	☐
JUSTIFICATION	
BY	
DISTRIBUTION /	
AVAILABILITY CODES	
DIST	AVAIL AND/OR SPECIAL
A	

DISTRIBUTION STAMP

DTIC
SELECTE
DEC 1 1982
D

DATE ACCESSIONED

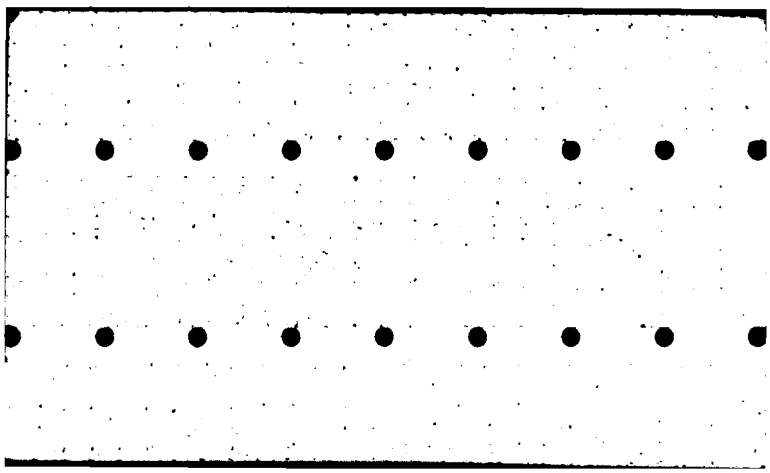
DTIC COPY INSPECTED 2

DATE RECEIVED IN DTIC

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-DDA-2

DTIC FORM OCT 79 70A

DOCUMENT PROCESSING SHEET



RIME:

The Recoverable Item Management Evaluator:

Volume I: Model Description

by

W. Steven Demmy

TR-80-01

DECISION SYSTEMS
2125 Crystal Marie Drive
Dayton, Ohio 45431

(513) 426-8515

160

AD-A124 984

WORKING PAPER
DO NOT PHOTOGRAPH

RIME:

The Recoverable Item Management Evaluator:

Volume I: Model Description

by

W. Steven Demmy

TR-80-01

DECISION SYSTEMS
2125 Crystal Marie Drive
Dayton, Ohio 45431

(513) 426-8515

1 DEC 1982

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) RIME; The Recoverable Item Management Evaluator; Volume I; Model Description		5. TYPE OF REPORT & PERIOD COVERED INTERIM August 79 - June 80
		6. PERFORMING ORG. REPORT NUMBER TR-80-01
7. AUTHOR(s) W. Steven Demmy		8. CONTRACT OR GRANT NUMBER(s) F33600-78-~~R-0314~~ C-0524
9. PERFORMING ORGANIZATION NAME AND ADDRESS Decision System 3575 Charlene Drive Dayton, Ohio 45432		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 23041A5
11. CONTROLLING OFFICE NAME AND ADDRESS 2750th ABW/PMA Bldg 1, Area C Wright-Patterson AFB, Ohio		12. REPORT DATE May 1980
		13. NUMBER OF PAGES 153
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

A

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Recoverable item, multi-echelon, inventory/repair, simulation, METRIC, MOD-METRIC, AFLC

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the Recoverable Item Management Evaluator (RIME), a FORTRAN simulation model for evaluating the relative cost-effectiveness of analytic optimization procedures proposed for use in Air Force Logistics Command recoverable item management systems. Major features of the model include (a) the use of actual Air Force demand histories to drive the model demand processes, (b) modeling of current Air Force statistical estimation procedures, and (c) modeling of the dynamic interations among initial provisioning, replenishment and distribution policies. Volume I describes major model concepts and features.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) RIME; The Recoverable Item Management Evaluator; Volume I; Model Description	5. TYPE OF REPORT & PERIOD COVERED INTERIM August 79 - June 80
	6. PERFORMING ORG. REPORT NUMBER TR-80-01
7. AUTHOR(s) W. Steven Demmy	8. CONTRACT OR GRANT NUMBER(s) F33600-78-~~R-0314~~ C-0524
9. PERFORMING ORGANIZATION NAME AND ADDRESS Decision System 3575 Charlene Drive Dayton, Ohio 45432	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 23041A5
11. CONTROLLING OFFICE NAME AND ADDRESS 2750th ABW/PMA Bldg 1, Area C Wright-Patterson AFB, Ohio	12. REPORT DATE May 1980
	13. NUMBER OF PAGES 153
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

A

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Recoverable item, multi-echelon, inventory/repair, simulation, METRIC, MOD-METRIC, AFLC

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the Recoverable Item Management Evaluator (RIME), a FORTRAN simulation model for evaluating the relative cost-effectiveness of analytic optimization procedures proposed for use in Air Force Logistics Command recoverable item management systems. Major features of the model include (a) the use of actual Air Force demand histories to drive the model demand processes, (b) modeling of current Air Force statistical estimation procedures, and (c) modeling of the dynamic interations among initial provisioning, replenishment and distribution policies. Volume I describes major model concepts and features.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Table of Contents

Chapter I

Introduction

The rapid developments in computer science and operations research techniques have spawned several technical revolutions in the past 30 years. Multi-echelon inventory theory is one of these areas. An 1950, only a few limited results from queuing theory were available to assist inventory managers. Today, computerized systems exist capable of computing jointly optimal inventory policies for thousands of items stocked at perhaps several hundred locations, considering assembly/sub-assembly relationships, and multi-echelon logistics support systems. Several of these models have become major components of Air Force Logistics Command (AFLC) data systems, and several other models have been proposed for implementation. Application areas covered by these systems include planning and budgeting, initial provisioning, replenishment, and distribution.

The Recoverable Item Management Evaluator (RIME) is a Fortran-based simulation model for evaluating the relative cost-effectiveness of analytic optimization procedures proposed for use in AFLC recoverable item management systems. Major features of RIME include:

1. Actual Air Force demand drives the model. The world is never stationary but changes with world events and with the introduction of new weapon systems. Hence, RIME uses actual Air Force demand histories to represent the demand processes and recoverable item flows of recoverable items. This is in contrast to the use of the a priori assumptions embedded in analytical multi-echelon inventory models.

2. Parameter values are based on statistical estimates. The parameters of underlying demand processes are never known with certainty, but must be estimated from statistical data describing past demands. In RIME, we utilize the same statistical estimation formulas used in Air Force data systems to determine demand rates, NRTS rates, and condemnation fractions. This information is recomputed on a quarterly basis, in a manner similar to that of the D041 Recoverable Item Requirements Computation System.

3. Dynamic system interactions are modeled. The world is not stationary, but changes as actual Air Force demand changes. These changes produce temporary imbalances in the Air Force supply system, and subsequent management actions to correct these imbalances. RIME simulates these effects. Consequently, RIME describes the dynamic period-by-period interactions which cannot be represented in the stationary assumptions of current analytical multi-echelon inventory models.

Of course, RIME is also only a model, and is thus only an approximation of the real world. However, we believe that RIME is a sufficiently rich description of Air Force supply systems to serve as a convenient test bed for comparing alternate analytical inventory management policies.

This report presents a detailed description of RIME. Volume I describes the major concepts, organization, and input/output features of the model, while Volume II provides detailed narratives of each RIME subroutine as well as listings of the FORTRAN source code. On the other hand, reference 1 describes the use of RIME in a study of the relative cost-effectiveness of several computation methods proposed for use in Air Force recoverable item management. Throughout this report, we assume the reader is familiar with Sections I-IV of Reference 1.

An Overview of the Recoverable Item Management Evaluator System

The major components of the Recoverable Item Management Evaluation System are shown in Figure I-1. At the heart of the system is the RIME Simulator. The Simulator simulates a multiple echelon recoverable item inventory system, and provides the means for estimating the procurement dollars and supply support measures associated with given sets of inventory management levels.

Figure I-1 illustrates the major components in the RIME system. As shown in the figure, the RIME System has four major components. The Data Extract System is used to obtain LRU/SRU data from the D041 Data Bank, and to manipulate this information into the form required for input to the RIME System. We call this data file the "Master Data Set." The Exogenous Event Generator translates the aggregate historical demands obtained from the Master Data Set into a detailed list describing LRU and SRU reparable generations and associated requisition, repair, and NRTS activities by base. These detailed lists are recorded on the Exogenous Event File, and represents a major input to the RIME Simulation Model.

The Stock Level Computation system also uses the Master Data Set as input. This system computes inventory levels for each time period in the simulation horizon for each item in the Master Data Set. By varying the input parameters, the system can compute stock levels using the METRIC, MOD-METRIC, Variable Safety Level (VSL), or AFLCR 57-27 computational methodologies, as well as several variations of these methods. As shown in the figure, different computational methods may be used in initial provisioning calculations from those used during the replenishment phase of an item's life cycle. The individual item stock levels computed by this system are stored in a "Levels" File. This File provides the second major input to the RIME Simulation Model.

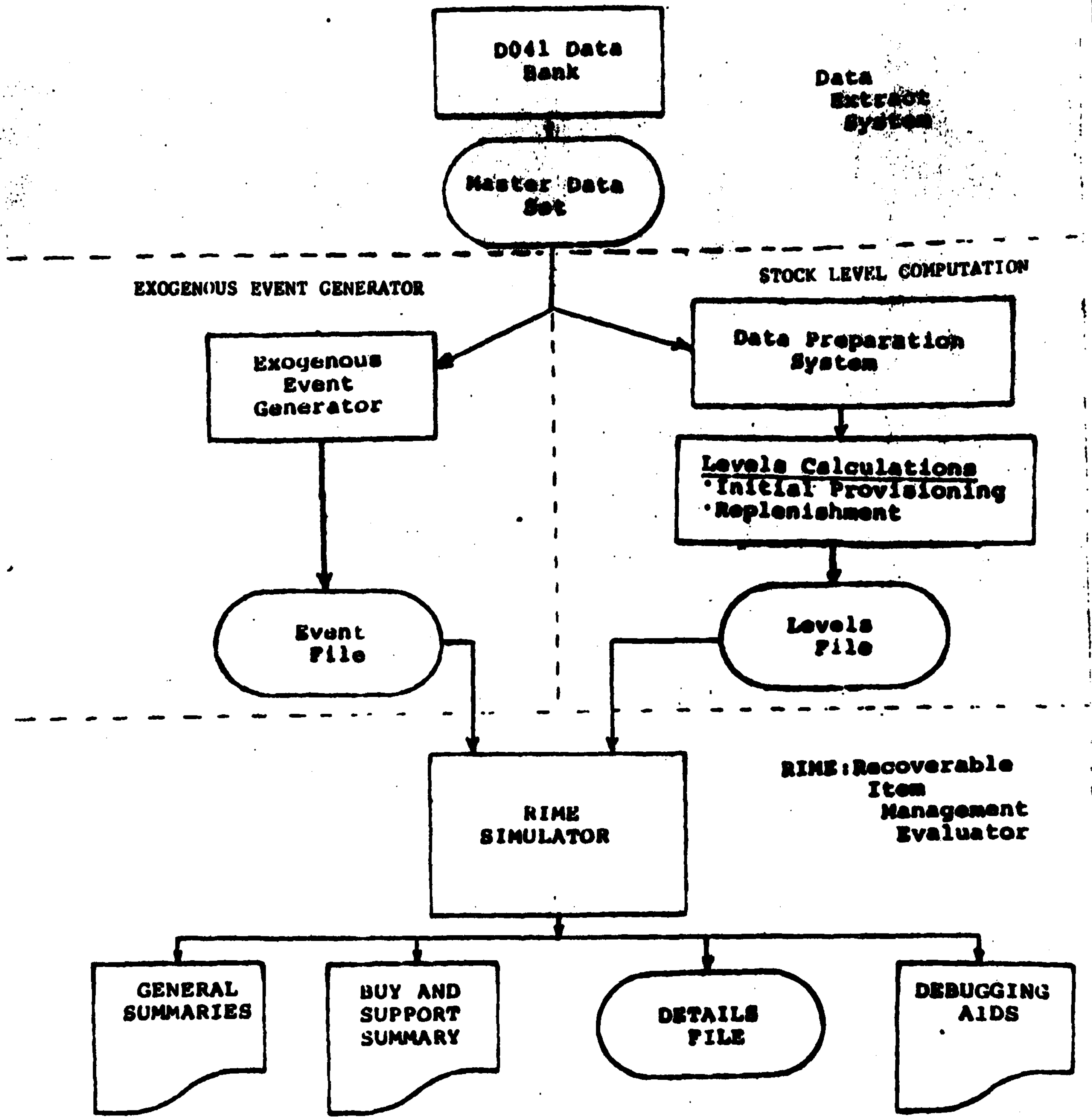


Figure I-1.

Major Components in RIME System.

The heart of the RIME system is, of course, the simulation model itself. This model inputs the detailed list of historical reparable generations and associated exogenous events, as well as sets of levels computed by the Stock Levels Computation system. It then uses simulation techniques to evaluate how well the given stock levels would have performed in managing the events described in the Event File.

Outputs from the simulation model include: (a) detailed statistics of inventory/repair activities by location, item type, and time period, (b) a short-form report which displays totals of six major statistics summed over all simulated periods, (c) results for each replication for each LRU/SRU group, and (d) printouts to assist in the debugging of extensions to the RIME model.

Chapters I through IV of Reference 1 provide a general overview of the RIME model, and we assume that the reader is familiar with this material. In this report, we provide detailed description of each of the components of the RIME system. In Chapter II, we review the major concepts employed in the RIME Simulation Model. This Chapter describes the numbering systems used to keep track of inventories at different stocking locations, methods of describing the passage of times within the simulated model, the specific events simulated by the RIME system, and a description of the detailed statistics collected by the Simulation Model. Chapter III provides a detailed description of the Events Generator. This Chapter describes methods for generating exogenous events and the computer programs used to implement the event generation activities. Chapter IV describes the Levels Computation System. This Chapter describes the network of computer programs involved in simulating specific combinations of initial provisioning and replenishment calculation methods. Finally, Chapter V provides instructions for individuals

interested in utilizing the current RIME version. This chapter describes the steps involved in simulating specific inventory management policies, and also describes output reports produced by the RIME Simulation Model.

Chapter II

Major Model Concepts

To understand the detailed program structure of RIME, one must possess a clear understanding of (a) the numbering system used to keep track of the stock status of each simulated LRU and SRU at each at the possible stocking locations, (b) the methods used to simulate the passage of time, (c) the events that simulate significant inventory system transactions, and (d) the data files that record current system status and that measure simulated performance. These major elements are the subject of this chapter.

Stock Keeping Unit Conventions

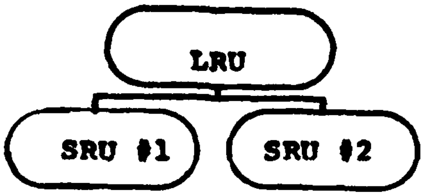
In RIME, each LRU and SRU may be stocked at possibly several stocking locations. As discussed in Chapter 1, the RIME model represents several types of stocking locations, including a depot level maintenance facility, several operating bases, and an aircraft overhaul facility. In simulating these systems, it is necessary to keep track of the on-hand and on order stock for each stock number at each of the possible locations. We use the term Stock Keeping Unit (SKU) to refer to the specific number assigned to a given stock number-geographic location pair.

Figure II-1 illustrates the assignment of Stock Keeping Unit Numbers for an LRU/SRU family consisting of an LRU and two SRUs that are stocked at three operating bases. As shown in the figure, inventories of the LRU at the Depot Level Maintenance Facility are always assigned a Stock Keeping Unit number of 1. Stock

FIGURE II-1.

STOCK-KEEPING UNIT NUMBERING CONVENTIONS
FOR
THREE BASES AND TWO SRUS

Sample Data Set



Consider a parts family consisting of an LRU and two SRUs. Suppose each item is stocked at three bases, plus an overhaul facility plus a Depot Level Maintenance Facility

Then NBASES = 3
NSRU = 2
NSTKU = (NSRU+1)*(NBASES+2)
= (3)*(5)
= 15

Stock Keeping Unit	Description
1	LRU at Depot Level Maintenance Facility
2	LRU at Base #1
3	LRU at Base #2
4	LRU at Base #3
5	LRU at Overhaul Facility
6	SRU #1 at Depot Maintenance Facility
7	SRU #1 at Base #1
8	SRU #1 at Base #2
9	SRU #1 at Base #3
10	SRU #1 at Overhaul Facility
11	SRU #2 at Depot Level Maintenance Facility
12	SRU #2 at Base #1
13	SRU #2 at Base #2
14	SRU #2 at Base #3
15	SRU #2 at Overhaul Facility

Keeping Unit numbers of 2, 3, and 4 are assigned to inventories of the LRU at base locations 1, 2, and 3 respectively. Finally, an SKU of 5 is assigned to inventories of the LRU at the Aircraft Overhaul Facility.

Stock Keeping Unit numbers for the SRUs follow a similar pattern. Consequently, an SKU of 6 represents inventories of SRU #1 at the Depot Maintenance Facility, while the SKUs of 7, 8, and 9 represents inventories of this SRU at Bases 1, 2, and 3, respectively. Finally, an SKU of 10 is assigned to inventories of SRU #1 at the Aircraft Overhaul Facility. As shown in Figure II-1, a similar pattern is used in assigning Stock Keeping Unit numbers to the inventories for SRU #2.

In RIME, the variable NSRU denotes the number of SRUs that are components of the LRU, and the variable NBASES denotes the total number of base locations. Hence, counting the LRU there are (NSRU+1) distinct stock numbers simulated in RIME, and each of these may be stocked at any of the NBASES bases in addition to being stocked at the depot and the overhaul facility. Hence, the total number of stock keeping units needed to keep track of inventories by location is (NSRU + 1)* (NBASES +2).

The Timing Mechanism

The fundamental modeling concept employed in RIME is that of an "event." An event is a specific point in time in which the state or condition of the system changes or potentially changes. For example, the amount of stock on hand changes if a requisition is received and goods are shipped to fill the demand. Hence, receipt of a customer requisition is an event. Other events that may change the amount of on hand stock include delivery of a replenishment order by a vendor of the supply system and the return of serviceable assets from a customer.

In simulating any system, two distinct categories of events may be identified. Exogenous events are events that are "caused by" activities that are external to the system; they are the stimuli that cause the system that react in some manner. On the other hand, endogenous events are caused by the reaction of the simulated system to either exogenous events or other endogenous events. For example, in RIME reparable generations of LRUs and SRUs and associated demands for serviceable replacements are treated as exogenous events -- events that drive the entire simulated repair/resupply system. The shipment of goods and the placement of replenishment orders are examples of endogeuous events. They are "caused by" the LRU and SRU demands which deplete on hand stocks.

In RIME, a number of events are treated as exogenous events; these are:

1. LRU and SRU reparable generations, and associated condemnation, serviceable return, repair, and NRTS actions whose timing may be determined from the timing of the reparable generation event.
2. The computation of initial provisioning stock levels.
3. The calculation of replenishment stock levels.

Because of the large volume of data involved in simulating an LRU and all of its SRUs simultaneously, all exogenous events for a given LRU/SRU family are generated as a preprocessing step and recorded on magnetic tape. This list of events is called the "Exogenous Event File (EEF)". Events recorded on this file are sorted in time sequence. Consequently, the time interrelationships of exogenous events may be determined by sequencially reading the EEF. On the other hand, the time interrelationships of Exogenous Events are maintained through the use of a Future Events List (FEL) that is updated dynamically as part of the RIME simulation process. We will discuss each of these files and illustrate their use later in this chapter.

RIME Events

In the Recoverable Item Management Evaluator, three categories of events are used to simulate recoverable item flows. They are:

a. Material flow events - describing the receipt of a customer requisition, delivery of replenishment order, the return of serviceable assets, the generation of unserviceable assets, and associated repair, condemnation, or NRTS actions.

b. Management action events - describing important management decision points. The calculation of control levels, status reviews to determine what reorder, termination, or disposal actions are appropriate, and the establishment of budgets and procurement guidelines are examples of these events.

c. Simulation bookkeeping events - events to collect time-valued statistics, to collect intermediate statistics for debugging purposes, and to signal the end of simulation.

At present, twenty possible events are included in the RIME model. These events are listed in Table II-1. In RIME, there is a separate FORTRAN subroutine to describe how each event changes the state of the system, and to record associated performance statistics. Table II-1 also presents the subroutines associated with each event and the definitions of the parameters IP3, IP4, and IP5 used in scheduling the respective events.

Event Parameters

In RIME, each event -- whether exogenous or endogenous -- is represented by five data elements. These are:

ITIME - The time the event is scheduled to occur.

ITYPE - The event type.

IP3, IP4, IP5 - Three parameters which further define the nature of the event.

Table II-1

RIME Events

General Variables

N = stock keeping unit number
IQTY = quantity associated with this event
KEND = week number within the current quarter

Event Type	Event	X=Exogenous N=Endogenous	Event Routine	Parameters IP3	IP4	IP5
1	Requisition received from a lower echelon.	X,N	REQ	N	IQTY	Priority[2]
2	Receipt of Replenishment Order	N	RECEIV	N	IQTY	NSKU[3]
3	Cancellation	X	CANCEL*	N	IQTY	
4	Serviceable Return	X	RET	N	IQTY	
5	Inv Status Review	N	REVIEW	N[1]	IFLAG[1]	
6	Levels Computation	X	LEVEL			
7	Buy Guidelines	X	GUIDE*			
8	Budget Review	X	BUDGET*			
9	Update History File	X	FORUPD*			
10	End of Run	N	OUTPUT			
11	Special Statistics	N	SSTAT	KEND	KEND	KEND
12	Endogenous Requisition Generation	N	DEMPAR*			
13	Trace	N	MAIN			

Notes

1 If N=0, review all SKU's. Otherwise, review item N only. If IFLAG=1, schedule all orders for immediate delivery to represent initial provisioning; if IFLAG=0, order using standard lead times.

2 Priroity=1 denotes a high priority requisition, while Priority=2 denotes a routine requisition. If the requisition is to provide parts for LRU repair, IP5=(100*NJOB+Priority). Hence, Priority = MOD (IP5,100). If the requisition is to resupply a second stock keeping unit (NSKU), then IP5 = 100*NSKU+Priority.

3 Source of resupply, where NSKU=0 denotes receipt of a vendor shipment.

* These events are not implemented in the current RIME model. However, these program names are reserved for future enhancements.

Table II-1 (Cont'd)

Event Type	Event	X=Exogenous N=Endogenous	Event Routine	Parameters IP3	IP4	IP5
14	Reparable Generation - record rep gen - increase WIP Note: For LRUs, IQTY must equal 1.	X	REPGEN	N	IQTY	NJOB
15	Condemnation - record condemnation - decrease WIP - review stock status	X	CONDEM	N	IQTY	NJOB
16	Begin Waiting for Parts - Update RTS statistics - file requirement for NNEED units	X	BWAIT	N	NNEED	NJOB
17	Receive Parts - reduce NNEED - If NNEED=0, schedule repair completion	N	RCVPRT	N	IQTY	NJOB
18	Repair completion - record completion - reduce WIP - increase serviceables - review stock status	N,X	CREPR	N	IQTY	NJOB
19	NRTS unit; begin transport to depot - record NRTS - decrease base WIP - increase depot WIP - review stock status	X	NRTS	N	IQTY	NJOB
20	Initial Provisioning - record buy - position stock	X	INPROV			

Notes

NJOB = Job number. Each reparable generation is numbered sequentially, beginning with NJOB=1001. The sequence numbers are assigned as events are created in the Events Generator, which is not necessarily the same order as the time sequence in which the events are scheduled.

NNEED = number of units needed (total overall SRUs components of the LRU) before NJOB may be completed.

As shown in Table II-1, event type 1 represents the receipt of a requisition from a customer of the supply organization. This event may be either exogenous (X) or endogenous (N), depending upon the origin of the requisition request. In simulating type 1 events, subroutine REQ is used. Further, the three parameters associated with a type 1 (requisition) event are defined as follows:

IP3 = The Stock Keeping Unit Number N associated with the inventory item being requisitioned.

IP4 = The number of units (IQTY) being requisitioned.

IP5 = The priority of the requisition. Definitions of the other event types and the associated FORTRAN routines may be seen by further inspection of Table II-1.

Timing Conventions and Initial Events

To simplify programming and analysis tasks, it was desirable to establish timing conventions that differ slightly from values associated with the Gregorian calendar. Specifically, RIME assumes that each day consists of 100 Time Units (TU). Days, weeks, months, and years are then assumed to be related as shown in Table II-2. Since the values are used repeatedly throughout a simulation run, specific COMMON variables are established for each of these time units. Values of these variables are initialized by subroutine INITAL at the beginning of each RIME simulation run. Other timing variables that are initalized by subroutine INITAL are shown in Table II-3.

All RIME events may be classified as either (a) scheduled events, (b) random events, or (c) dependent events. Scheduled events are events with a specific

TABLE II-2

Time Conventions

Time Intervals	Basic Value	Time in TU's		Variable
1 Day		100 TU's	=	ITDAY
1 Week	7 Days	700 TU's	=	ITWEEK
1 Month	4 Weeks	2,800 TU's	=	ITMNTH
1 Quarter	3 Months	8,400 TU's	=	ITQTR
1 Year	4 Quarters	33,600 TU's	=	ITYEAR

NOTES

It is assumed that 1 Day = 100 Time Units (TU's), and each Time Variable is expressed in TU's.

TABLE II-3

Timing Vriables Established in INITAL

Variable	Definition
IQTRND	clock time that marks end of current quarter
ITINV	number of current quarter
ITIME	current simulation clock time (100 Time Units = 1 day). At the beginning of a simulation run, ITIME = 0.
ITLEVL	clock time of the next computation of stock levels
IDLEVL	time interval between stock level computation
ITDIV*	clock time of the next division level review
IDDIV*	time interval between division level reviews
ITFOR*	time of the next update of demand history records (Event type 9)
IDFOR*	time between history file updates
ITHQ*	time of next Headquarters USAF stock fund budget review
IDTHQ*	time between Headquarters USAF budget reviews.
ISTOP	clock time that simulation is to be stopped
ISTAT	clock time for activating statistics collection routines
IDSTAT	time interval between statistics collection
INQTR	number of quarters to be simulated

NOTES

*These variables are not used in the current verision of RIME. However, these variable names are reserved for future extensions of the model.

schedule of occurrence times. For example, initial provisioning calculations (event type 20) occur at the beginning of each simulation run, and leveling events (event type 6) occur at the end of every quarter. On the other hand, reparable generations (event type 14) and serviceable returns (event type 4) are random events--the time of occurrence of these events is not known in advance. Finally, the delivery of a replenishment order (event type 2) is a dependent event since the time of delivery is dependent upon earlier management reordering decisions. Dependent events are caused by the occurrence of an associated random or scheduled event. Hence, requisitions for serviceable assets (event type 1) to replace flight line reparable generations are dependent events, since the requisition for a serviceable unit was caused by the flight line failure.

In RIME, the first occurrence of each scheduled event is created by subroutine INITAL; that is, INITAL puts an entry on the Future Events List for each scheduled event at the beginning of a simulation run. Subsequent scheduled events are put on the Future Events List each time a new scheduled event occurs. For example, at the conclusion of a Leveling event (event type 6), a subsequent type 6 event is scheduled to occur one quarter in the future.

Specific values for the time between scheduled events are set in subroutine INITAL. Table II-3 defines specific time variables initialized in this routine.

The Future Events List

The sequencing of endogenous events through time is controlled by a Future Events List (FEL). The FEL is a list of all endogenous events scheduled to occur at some future (simulated) time. For each event on the list, the following values are recorded.

ITIME = The time the event is scheduled to occur.

ITYPE = The event type (code).

IP3, IP4, IP5 = Event parameters.

Structurally, the FEL is a linked list in ascending order by scheduled event time. In such a list, new entries to the list are inserted in a previously unused data storage location, and pointers are used to indicate the correct sequencing of the entry in the list. The major variables associated with this list are shown in Table II-4.

At the beginning of each simulation run, subroutine INFEL is called to initialize the pointers and other control variables associated with the FEL. Scheduled events are then entered and removed from the Future Events List by subroutine ENTER and REMOVE, respectively. New events are inserted into the FEL by the following CALL statement:

CALL ENTER (ITIME, ITYPE, IP3, IP4, IP5)

where ITIME, ITYPE, IP3, IP4, IP5, are as defined above. When subroutine ENTER is called, the new event data is recorded, and the list pointers are updated to insert the new event in the proper time sequence.

Events are removed from the Future Events List by subroutine REMOVE; the CALL statement takes the form

CALL REMOVE (ITIME, ITYPE, IP3, IP4, IP5)

and the calling parameters are as defined above. Subroutine REMOVE removes the <u>first</u> entry from the FEL and updates the pointers accordingly; that is, it deletes the list entry with the smallest (earliest) scheduled clock time from the FEL, and returns the corresponding data values through the parameter list.

TABLE II-4

Future Events List Variables

NFEMAX	=	maximum number of entries on Future Events List
NENTRY	=	number of entries on the Future Events list
ILOCFE(J)	=	number of Jth unused data location in the Future Events file
K	=	list index
JTIME(K)	=	clock time of event transaction
JTYPE(K)	=	event type
JPOINT(K)	=	pointer to next transaction in list
JFSN(K)	=	stock number identification*
JQTY(K)	=	quantity involved*
JPRIOR(K)	=	priority,* where

$$JPRIOR = \begin{cases} \text{1-high priority} \\ \text{2-low priority} \end{cases}$$

NFIRST	=	location of first transaction of chain
NTIME	=	time of earliest transaction on list
NLOC	=	location of last transaction placed on the chain

*If applicable

Subroutine WRIFEL is another routine associated with the Future Events List. In debugging new RIME models, it is sometimes desirable to write out the entire Future Events List at selected times within a simulation run. Subroutine WRIFEL performs this function.

To illustrate the use of the Future Events List consider the LRU family illustrated in Figure II-1. Suppose that as a result of an LRU failure at base #2, a high priority requisition for a serviceable replacement is submitted to the base supply organization. In this case,

ITYPE = 1 (The Event Code for a requisition event, as will be discussed later).

IP3 = 3 (The stock-keeping unit number for LRU inventories at base #2).

IP4 = 1 (The number of units being requisitioned).

IP5 = 1 (This indicates a high priority requisition. For requisition events, IP5 = 1 denotes a high priority requisition, while IP5 = 2 denotes a low priority requisition. For other event codes, this parameter has other meanings.)

Suppose this requisition is to occur at the beginning of the 9th day of the simulation. Then using the time conventions illustrated in Table II-2,

ITIME = 900

since there are 100 Time Units in each simulated day, and this event is to occur at the beginning of the ninth day.

This particular event may then be scheduled (i.e., put on the Future Events List) by setting each of the above variables as shown, and then using the FORTRAN call statement:

```
CALL ENTER (ITIME, ITYPE, IP3, IP4, IP5).
```

When the simulated clock time eventually reaches 900 time units, the above event data would be removed from the FEL and an appropriate subroutine would be called to simulate the actions which result from receipt of this requisition.

Subroutine EVNTS: The Event Scheduler

The scheduling of specific exogenous and endogenous events is controlled by Subroutine EVNTS. The general logic of this routine are illustrated in Figure II-2. As shown in the figure, subroutine EVNTS first initializes the pseudo-random number generator and the Future Events List, and then reads in data describing the LRU/SRU family currently being simulated. This is done by calls to subroutines RANDU, INITAL and INITM2, respectively. The routine then inspects the most imminent events (that is, the events with the lowest scheduled clock time) on both the Exogenous Event File (EEF) and the Future Event List (FEL). It then removes the event with the lowest scheduled clock time from the appropriate list , and calls the subroutine associated with that event. For example, if the event with the lowest scheduled time is a type 1 event, subroutine REQ is called. The routines associated with each event type are defined in Table II-1. These event routines update the status of the system and work-in-process inventories to reflect occurrence of the event. In addition, the event routines update appropriate performance statistics. Some of the events also cause additional endogenous events to be scheduled. If this occurs, subroutine ENTER is called to place the newly-created event on the FEL. For example, when a stock level computation event (Event Type 6) occurs, subroutine LEVEL is called to compute new stock levels for each LRU and SRU at each stocking location. At the conclusion of its calculations, subroutine LEVEL calls subroutine ENTER to schedule the next stock level computation event.

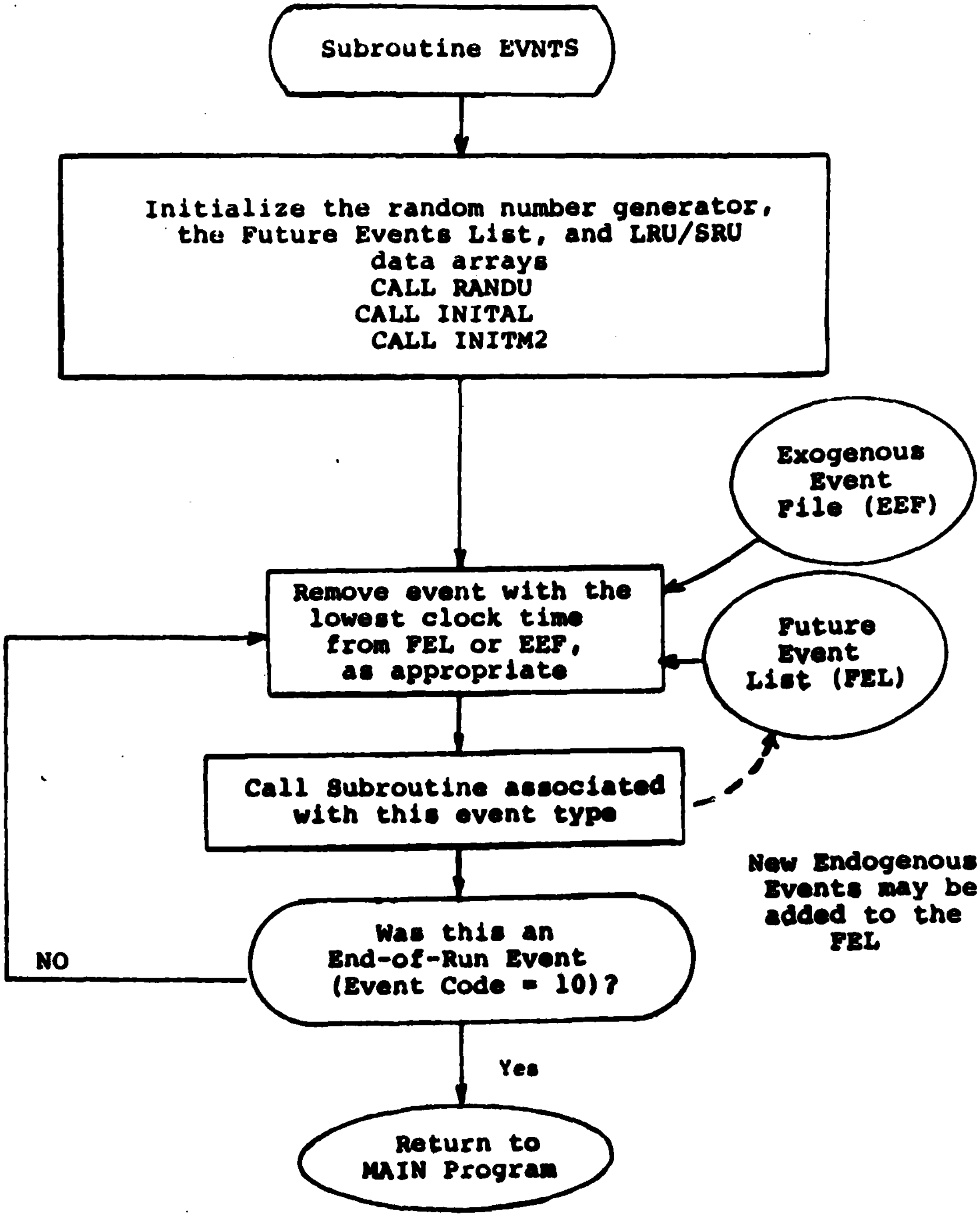


Figure II-2. Event Scheduling in Subroutine EVNTS

When an End-of-Run Event (Event Code 10) is encountered, Subroutine EVNTS returns to the MAIN program. Otherwise, the routine again scans both the exogenous event file and the updated FEL, and identifies the particular event with the lowest clock time. This event is then removed from the appropriate list and the process continues until a End-of-Run Event is eventually encountered.

Data Files

A large number of data elements are required to operate any computer simulation model, and RIME is no exception. The major categories of data associated with RIME are illustrated in Figure II-3. As shown in the Figure, RIME data elements may be logically classified into one of the following files:

1. Exogenous Event File (EEF).
2. Future Events List (FEL).
3. System Characteristics File.
4. Waiting-for-Parts File.
5. Item Data File.
6. Stock Level File.
7. Backorder File.
8. Simulation Counters and Flags File.
9. Performance Statistics File.

The Exogenous Event File and Future Events List have been discussed earlier. Let us now consider the contents of each of the remaining data files.

System Characteristics File

The Systems Characteristics File contains data elements that define the system as a whole. The set of items to be included in a RIME simulation run, the number of operating bases to be simulated, and the number of historical data periods available are examples of data elements in this file. This information is provided primarily through inputs of file 05. Table II-5 defines the major variables contained in this logical data file. Specific data elements input through this file will be discussed in detail in Section VI.

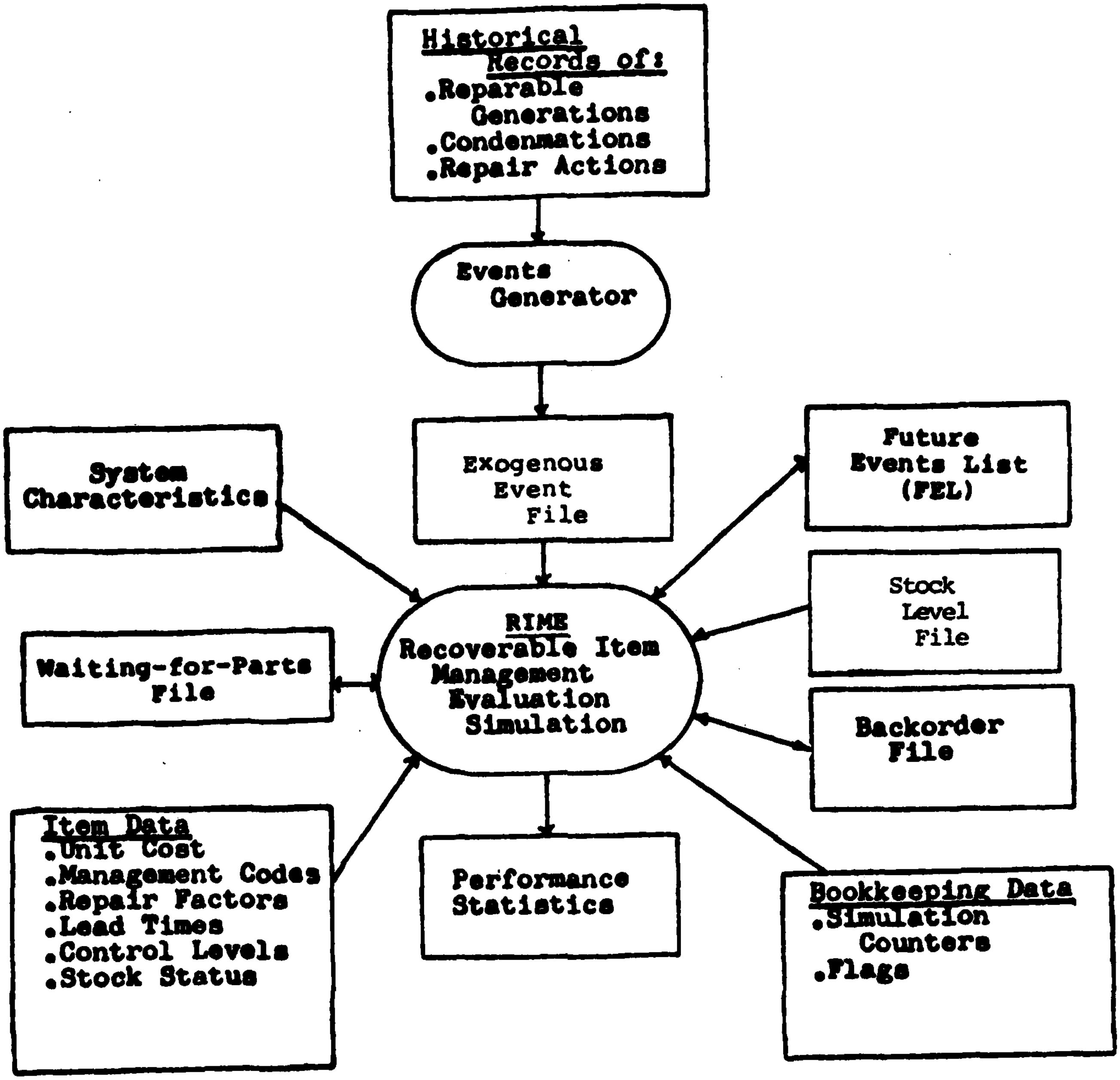


FIGURE II-3
Logical Data Files in RIME.

TABLE II-5

System Characteristics File

NJOB	Last assigned job number. NJOB is initially set to 1001, and is incremented each time an exogenous event is created.
NBASES	Number of bases, not counting the depot or the overhaul facility. Hence, the total number of stocking locations equals NBASES+2.
NSRU	Number of SRU's in the LRU
NITEM	Number of stock keeping units, which equals (NBASES+2)* (NSRU+1)

Work-In-Process File

To repair an unserviceable LRU, it may be necessary to remove and replace possibly several of the LRU's components. When this situation occurs, a "Wait-for-Parts" event (event type = 16) is scheduled, and an appropriate entry is made to the Work-In-Process File. Table II-6 illustrates the data elements contained in this file.

The GASP Simulation routines FILEM and RMOVE are used to enter and remove entries in the Work-In-Process File, and the Subroutine INGASP is used to initialize the appropirate GASP file variables. INGASP is called by Subroutine INITAL at the beginning of each RIME simulation replication.

A detailed description of the GASP file system and its associated routines is presented by A.A.B. Pritsker (The GASP IV Simulation Language, John Wiley and Sons, New York, 1974), and will not be discussed further in this report.

Item Data File

The item data file contains information specifically related to each item being simulated. An item's unit cost, its current level of on-hand, on-order and work-in-process stocks, and its control levels (e.g., reorder level, support level, termination level, or retention level) are examples of this type of data. Table II-7 presents a detailed list of information items which are contained in this file. As shown in Table II-7, the item data file contains several subcategories of information. These define (a) specific physical and management characteristics of the item, (b) data describing the stock status of the item, (c) variables describing repair and shipping times for the item, (d) variables describing the demand history for the item, (e) forecast variables which quantify forecast usage rates and demand variability and (f) control levels used in the management of the item's inventories.

TABLE II-6

Work-In-Process File

Data Element	Definition
N	The Stock-Keeping-Unit number of the LRU that is waiting for parts
NNEED	The total number of SRU components that are required before repair of the LRU can be commenced. For example, if the LRU requires three units of SRU#4 and one unit of SRU#6, then NNEED = 3+1 = 4.
NJOB	The job number associated with this reparable generation
ITIME	The simulated clock time that is entry was placed in the Wait-for-Parts File

TABLE II-7

Item Data File

Item Characteristics

Variable	Definition
N	Item Index
NITEM	Number of items simulated
ALC	Air Logistics Center that manages this item
FSN	Federal Stock Number, where FSN (1) = Material Management Code FSN(2) = Federal Stock Class FSN(3) = First six characters of NIIN FSN(4) = Remaining characters of NIIN
UM	Unit of measure
NOUN	Item name
MGTCD	Item Management Codes, where MGTCD(1) = Essentiality code MGTCD(2) = Supply management grouping MGTCD(3) = Weapon system code MGTCD(4) = Type computation code
ISEQ(N)	Sequence number for SKU N
UCOST(N)	Unit cost (dollars)
LTADM(N)	Administrative lead time (days). Time required to process and transmit a requisition from a stocking location to the resupply location.
LTPROD(N)	Production Lead time (days). Time required to manufacture (if applicable), package, and transport a resupply order for delivery of stock keeping unit N. This time does not include time spent waiting for parts. If N is a base level stock keeping unit, LTPROD(N) denotes the depot-to-base ship time.

Item Stock Status

INWIP(N)	Units in repair cycle (work-in-process)
INVACT(N)	Actual serviceable inventory on hand (units)
IVDUE(N)	Inventory due-in from supplier (units)
NBOIU(N)	Priority I backorders, in units
NBOIR(N)	Priority I requisitions backordered
NBOTU(N)	Total backorders, all priorities, in units
NORDPT(N)*	Pointer to most recent outstanding order (Note: This variable is not used or updated in the current RIME model)
NBOPT(N)*	Pointer to oldest, highest priority backorder

***Note**

By convention, pointers are set to zero if there are no other elements in the data chain.

TABLE II-7 (Cont'd)

Reparable Item Data

Variable	Definition
IBDTT(N)	Time to ship from base to depot (days)
IBRT(N)	Base Repair Time (days)
IDRT(N)	Depot Repair Time (days)
IDORT(N)	Depot Overhaul Repair Time (days)

History Variables

NDHIS	Number of periods saved in history file
NDENT(N)*	Number of forecast periods since item N enetered system (negative values indicate item has not yet entered system)
NDEMAC(N)*	Counter to accumulate demand for Nth item during the current forecast update period.
NDEMND(N,I)*	Demand for item N during Ith past period, I=1, 2,...,NDHIS
NREQAC(N)*	Counter to accumulate the number of requisitions for item N during the current period.
NREQ(N,I)*	Number of requisitions for item N during Ith past period.

Forecast Variables

ADR(N)*	Net annual demand rate for item N; the rate of unit demands less serviceable returns.
RMEAN(N)*	Estimate of mean demand rate during next demand period.
RTREND(N)*	Estimate of trend in demand per period.
RMAD(N)*	Mean Absolute Deviation of unit demand per period.
PERSUM(N)*	Sum of past forecast errors.
KNT(N)*	Counter used for adaptive smoothing forecasts.
RSIGLT(N)*	Estimate of standard deviation of lead time demand.
REQSIZ(N)*	Estimate of mean requisition size for item N.
REQMAD(N)*	Mean absolute deviation of requisition size for item N.

Control Levels

IROL(N)	Reorder level for item N
IRQTY(N)	Reorder quantity
ISUL(N)	Support level
IRL(N)	Retention level
ITL(N)	Termination level

Notes

*These variable are not used in the current implementation of RIME. However, these variable names are reserved for future extensions of the model.

In the current implementation of RIME, forecast and levels computation events are treated as exogenous variables, and all forecasting and stock level calculations are performed in the Levels Computation module as a preprocessing step. Consequently, the History and Forecast Variables defined in Table II-7 are neither used nor updated in the current RIME model. However, these data names are reserved for future use in studies in which these activities are to be treated as endogenous events.

Stock Level File

As noted in the previous paragraph, stock levels computations are performed as a pre-processing step in the Levels Computation module. These pre-computed levels are recorded in the Stock Levels File. When a Levels Computation Event (Event Code 6) occurs in RIME, subroutine LEVEL reads the Stock Levels File to determine the required stock levels for each Stock Keeping Unit. See Chapter IV, "The Levels Computation System", for a discussion of the data elements in this file.

Backorder Files

The backorder file records all outstanding backorders in the inventory system at a given point in time. This file contains a record for each requisition in a backorder status. Specific data elements in this file are defined in Table II-8. Included in these data elements are the SKU number associated with the backorder, the backorder quantity and priority, and the time the item was backordered. A pointer system is used to relate entries in the backorder file with the specific stock-keeping-units associated with that backorder. These pointer variables are also defined in Table II-8.

TABLE II-8

Backorder File

Variable Name	Definition
J	File Index
ITMBAC(J)	simulated clock time that the Jth entry in the backorder file was placed into the file
IDFSNB(J)	item number associated with the Jth backorder. If IDFSNB(J)=J, the backorder is for an independent demand item. Otherwise, if IDFSNB(J) is less than 1000, the backorder originated from stock-keeping unit IDFSNB(J). If IDFSNB(J) is greater than 1000, the backorder is a parts requirement for the repair of LRU reparable generation IDFSNB(J).
IPRIOR(J)	priority of Jth backorder
IQTYB(J)	quantity backordered
IBACPT(J)	pointer to the file entry of the next backorder for this same item. If IBACPT(J) = 0, there are no more backorders in the list for this SKU.
NBMAX*	maximum number of entries in backorder file
ILOCBK(K)*	index of Kth unused data location in backorder file
NLOCBK*	number of unused data locations.

NOTES

*NBMAX, ILOCBK, and NLOCBK are initialized by subroutine INITAL. Backorders are placed in this file by subroutine ENTERB, and removed from the file by subroutine FILLBO.

Performance Statistics File

The Performance Statistics File contains measures of the levels of activity observed during a simulation run. Table II-9 defines the specific data elements contained in this file. As illustrated in the Table, each performance measure has three different indexes, denoted by the variables I, J, and K. The variable I denotes a period index and represents quarter the during the simulation in which the statistic was observed. The variable J denotes the unit of measurement associated with the statistic. Values of J are:

$$J = \begin{cases} 1 = \text{a count of actions for Federal Stock Numbers affected.} \\ 2 = \text{the total number of units affected.} \\ 3 = \text{the dollar associated with all of these units} \end{cases}$$

Finally, the variable K denotes an aggregation category. Specific values utilized in RIME are as follows:

$$K = \begin{cases} 1 = \text{LRU at a Base.} \\ 2 = \text{SRU at a Base.} \\ 3 = \text{LRU at the Depot.} \\ 4 = \text{SRU at the Depot.} \\ 5 = \text{LRU at the Aircraft Overhaul Facility.} \\ 6 = \text{SRU at the Aircraft Overhaul Facility.} \end{cases}$$

For example, suppose that the reparable generation of two units of an SRU occurs at Base #3 during the eighth quarter of the simulation. To reflect this event, we must update the Performance Statistic IREPGN (I, J, K). Since the rep gen occurs in quarter 8, the period index is I = 8. Also, since we are dealing with an SRU at base level, the aggregation index is K = 2. Since we observed a single reparable

TABLE II-9

Performance Statistics File

Variable		Definition
	I	Period index, where I= 1, 2, ..., 16
	J	Type of measure, where J = {1 = actions/FSN; 2 = units; 3 = dollars}
	K	Aggregation cateogry, where K = {1 = LRU at Base; 2 = SRU at Base; 3 = LRU at Depot; 4 = SRU at Depot; 5 = SRU at Overhaul Facility; 6 = SRU at Overhaul Facility}
1.	INVOH (I, J, K)	inventory on hand at end of period
2.	INVOR (I, J, K)	inventory on-order at end of period
3.	IRECET (I, J, K)	receipts
4.	IRETRN (I, J, K)	returns
5.	INVDAY (I, J, K)	inventory-weeks
6.	IORDER (I, J, K)	orders placed
7.	IDISPS (I, J, K)*	disposals
8.	ITERM (I, J, K)*	terminations completed
9.	IEXPED (I, J, K)*	expediting actions
10.	IRATON (I, J, K)*	rationing actions
11.	IREQC (I, J, K)*	total requisitions cancelled
12.	IREQT (I, J, K)	total requisitions received from customers
13.	IREQI (I, J, K)	priority I requisitions received from customers
14.	IBACKT (I, J, K)	total backorders (end of period)

Note: *These variables are not used in the current version of RIME.

TABLE II-9 (Cont'd)

Variable		Definition
15.	IBACKI (I, J, K)	priority I backorders (end of period)
16.	IBAKDT (I, J, K)	total backorder weeks within period (Note: J=1 denotes requisition weeks of backorders See SSTAT)
17.	IBAKDI (I, J, K)	priority I backorder weeks within period (Note: J=1 denotes requisition weeks of backorders See SSTAT)
	IBODAT (I, J, K)	total time on backorder for backorders filled in period I (time is in TUs, where 100 TUs = 1 day)
	IBODAI (I, J, K)	total time on backorder for Priority 1 backorders filled in period I (time is in TUs)
18.	IFILLT (I, J, K)	total fills (number of requisitions filled immediately upon receipt)
19.	IFILLI (I, J, K)	priority 1 fills
20.	ISHIPT (I, J, K)	total shipments
21.	ISHIPI (I, J, K)	total priority I shipments
22.	ISMORD (I, J, K)*	total small orders (i.e., dollar value of order is less than a critical value specified as input.
23.	ILGORD (I, J, K)*	total large orders
24.	IBOPOH (J)	on hand inventory at beginning of simulation
25.	IBOPOR (J)	on-order inventory at beginning of simulation
26.	IBOPBO (J)	backorders at beginning of simulation

Repair Statistics

1.	IREPGN (I, J, K)	reparable generations at this location (for depot, NRTS items are not inlcuded.)
2.	INRTS (I, J, K)	number of units not reparable at this station (NRTS). For a base, these are NRTS units sent to the depot. For depot locations (i.e., for K=3 or 4) this variable represents NRTS assets routed to the depot.

TABLE II-9 (Cont'd)

Repair Statistics

3.	ICNDEM (I, J, K)	number of reparable generations condemned this period. Note: Number of NRTS items = IREPGN-IRTS-ICONDM
4.	IRECPL (I, J, K)	number of repairs completed
5.	IWIP (I, J, K)	Work in process at end of period
6.	IWFP (I, J, K)	Time waiting for parts - (in TUs, 100 TUs = 1 day) (Computed by RCVPRT when parts are finally received, in the quarter received.)

generation, the action count (J = 1) statistic IREPGN (8, 1, 2) is increased by 1 even though two assets were involved. However, since two assets generated at this time, the units count (J = 2) statistic is increased from IREPGN (8, 2, 2) to a value of IREPGEN (8, 2, 2) + 2. Finally, suppose that the LRU has a unit price of $6,500. In this case, the dollars (J=3) statistic IREPGN (8, 3, 2) is increased by 2 x $6,5000 = $13,000. The subroutine CUM is used to perform these update calculations for all of the Performance Statistics shown in Table II-8.

Simulation Counters and Flags

Every simulation model requires a series of counters and flags to control the progress of the simulation, and to perform necessary bookkeeping tasks. The number of quarters to be simulated, the number of the current statistics collection interval, and the number of simulation runs to be performed are examples of these types of data elements. The variables of the Future Events List are an important example of this type of information. Other major variables in this category are defined as inputs to RIME through File 05. These latter variables are discussed in detail in Section VI.

Chapter III

The Events Generator

The Events Generator uses Monte Carlo techniques to generate detailed lists of LRU and SRU reparable generations and all related requisition, condemnation, and repair activities. This chapter describes the major features of this system and the computer codes which implement it.

The Event Generation Process

A major design feature of the RIME evaluation model is that all recoverable item flows are driven by actual Air Force histories of recoverable item activity. For example, Table III-1 presents historical data from the D041 Depot Data Bank for Federal Stock Number 5841-00-2475357. This FSN is a Line Replaceable Unit for the F-111 aircraft with a price of $6,150 per unit. This LRU has a single SRU component; the SRU is FSN 5841-00-4899855, a SYNC ASSY with a unit price of $452.20. Table III-1 presents the repairable item generation data for these two stock numbers for the first quarter of FY74, as well as the total repairable generation in each category shown for the four year period from FY74-1 through FY78 -4.

As shown in the table, this LRU had 115 repairable generations during the first quarter of FY74. Of this total, 110 were repaired at base level, while 5 were NRTS assets. That is, five assets were returned to the depot for repair. None of the total of 115 assets were condemned at the base level, and there were no depot repairable generations of this LRU during FY74-1 (i.e., there were no generations

Table III-1
D041 Reparable Generation Data
For FSN 5841-00-2475357
An F-111 LRU

Noun: SYNCHRO134
Price: $6,150

Data Element	Description	Code	Quarter FY74-1		Totals for FY74-1 thru FY78-4	
			LRU	SRU	LRU	SRU
1.	Reparable Generations	BRGN	115	10	1980	124
2.	Assets Repaired at This Station	RTS	110	2	1945	10
3.	Base Condemnations	CON	0	0	0	1
4.	*Assets Not Reparable at This Station*	NRTS	5	8	35	113
5.	Depot Reparable Generations	DBGN	0	0	0	2
6.	Depot Condemned	DCND	0	0	0	0
7.	Depot Repaired*	DREP*	5*	10*	50*	103*
8.	Depot Overhaul Condemned	OVCN	0	0	0	0
9.	Installed Program (100's hours)	PROG	300	300	4248	4248

* The total number of assets which completed repair in a given quarter is available from the D041 Depot Data Bank. However, this information is not used in RIME, since these numbers reflect the performance of past inventory management policies, and that are *not* necessarily representative of the inventory management rules to be evaluated.

Table III-1 (Cont.)

SAMPLE HISTORICAL DATA

FSN	NOUN	PRICE
5841002475357	SYNCHRO134	6150.00

LRU

	1	2	3	4	5	6	7	8
BRGN	115	131	102	103	126	111	138	145
RTS	110	131	102	101	120	109	134	137
BCON	0	0	0	0	0	0	0	0
NRTS	5	0	0	2	6	2	4	8
DRGN	0	0	0	0	0	0	0	0
DCND	0	0	0	0	0	0	0	0
DREP	5	4	0	0	1	1	1	6
OVCN	0	0	0	0	0	0	0	0
PROG	300	278	270	304	304	262	288	230

	9	10	11	12	13	14	15	16	Total
BRGN	135	177	146	111	107	134	85	112**	1980
RTS	135	176	146	109	107	132	85	111**	1945
BCON	0	0	0	0	0	0	0	0**	0
NRTS	0	1	2	2	0	2	0	1**	35
DRGN	0	0	0	0	0	0	0	0**	0
DCND	0	0	0	0	0	0	0	0**	0
DREP	6	10	5	5	1	2	1	2**	50
OVCN	0	0	0	0	0	0	0	0**	0
PROG	236	258	303	266	289	269	189	202**	4246

FSN	NOUN	PRICE
5841004899855	SYNC ASSY	452.20

SRU

	1	2	3	4	5	6	7	8
BRGN	10	14	2	12	8	7	13	6
RTS	2	2	0	0	1	2	0	0
BCON	0	0	0	0	0	0	0	0
NRTS	8	12	2	12	7	5	13	6
DRGN	0	0	0	0	0	0	0	0
DCND	0	0	0	0	0	0	0	0
DREP	10	1	6	10	7	3	7	9
OVCN	0	0	0	0	0	0	0	0
PROG	300	278	270	304	304	262	288	230

	9	10	11	12	13	14	15	16	Total
BRGN	8	5	16	8	6	3	1	5**	124
RTS	2	0	1	0	0	0	0	0**	10
BCON	0	1	0	0	0	0	0	0**	1
NRTS	6	4	15	8	6	3	1	5**	113
DRGN	0	0	0	2	0	0	0	0**	2
DCND	0	0	0	0	0	0	0	0**	0
DREP	5	6	8	7	12	7	2	3**	103
OVCN	0	0	0	0	0	0	0	0**	0
PROG	236	258	303	266	289	269	189	202**	4246

from the Overhaul Facility). During this quarter, the LRU had a total installed program of 30,000 hours. Similarly, the SRU had 10 repairable generations during this quarter. Two of these were repaired at base level, while 8 were returned to the depot.

In simulating this LRU/SRU family for the quarter FY74-1, exactly 115 repairable generations of the LRU are created in the RIME model, with exactly 110 of these to be repaired at base level. Similarly, exactly 10 SRU repairable generations are created in the simulation of this quarter, with 2 repaired at base level and 8 returned to the depot. In a simulation of the entire FY74-1 through FY78-4 interval, exactly 1980 LRU generations would be created, and exactly 124 SRU generations. Similarly, all of the other historical data displayed in Table III-1 would be exactly reproduced by the simulation model.

One problem in simulating detailed repairable item flows from Air Force historical data is that the historical data is maintained in an aggregated form. That is, the historical records only tell us the total number of repairable generations that occurred at all base locations during a given interval, and provides no information in terms of which specific bases generated these failures. Similarly, the historical records provide no data which allow us to link up specific SRU repairable generations with associated LRu failures. Consequently, it was necessary to devise probability models to interrelate LRU and SRU reparable generations. Table III-2 presents the assumptions utilized in our demand generation process. Basically, the rules presented in this table are based on a relatively small number of fundamental assumptions. These are (a) Air Force D041 recoverable item flow histories are to be reproduced as closely as possible in the simulation process, (b) the probability that a specific SRU failure in a given quarter is related

Table III-2

Basic Assumptions of RIME Reparable Generation Probability Model

I. Base Reparable Generations

1. The probability that a specific LRU reparable generation occurs at a given location is proportional to the flying activity at that location relative to the total flying activity at all locations in the specific quarter under consideration. Further, it is assumed that a uniform distribution describes the probability that a given LRU rep gen occurs at any specific instance within the quarter under consideration. This is equivalent to assuming that LRU reparable generations follow a simple Poisson process within the specific quarter of interest. However, the exact number of LRU reparable generations within a given quarter exactly equals the historical values recorded in D041.

2. The probability that a given SRU rep gen (RTS, NRTS, or condemnation) is related to a given LRU rep gen is equal to the ratio of the total SRU rep gens in a given quarter to the total number of SRUs installed in LRUs that fail during that quarter. We refer to this as the Exposure Probability model. Once an SRU rep gen is related to a specific LRU rep gen, the clock times for related SRU events are determined by adding appropriate time delays to the LRU failure time.

3. If recorded D041 LRU rep gens exceed recorded D041 SRU rep gens, it is assumed that some LRUs were repaired without requiring replacement SRUs. Calibration and adjustment actions and job-routed repairs are examples of this situation.

4. If recorded D041 SRU rep gens exceed the total SRUs installed in failing LRUs, we assume the excess units are "independent SRU demands"; that is, demands that are independent of an associated LRU turn-in to base supply. This situation will occur if an LRU is repaired at the flight line, rather that in the base maintenance shops.

5. If an LRU is condemned, all SRUs in the condemned LRU that are not reparable or condemned are treated as serviceable returns to the supply system.

II. Depot Reparable Generations

1. LRU depot delays are simulated using the same assumptions employed in the METRIC and MOD-METRIC models; namely, LRU depot delays are treated as independent random variables, independent of SRU stock status at the depot. Hence, although the LRU repair time may include an allowance for parts delays, these delays are not explicitly simulated.

2. Since all LRU depot delays are treated as independent random variables, all SRU depot reparable generations are also treated as "independent"; that is, these generations are not related to any of the LRU generations.

Table III-2 (Cont.)

3. It is assumed that the specific time that a given LRU or SRU depot reparable generation occurs is uniformly distributed over the specific quarter under consideration. This is equivalent to assuming that both LRU and SRU depot rep gens obey simple Poisson processes.

III. Forecasting Assumptions

1. All values for Mean Time Between Demands (MTBD), NRTS rates, and condemnation rates are based upon eight-quarter moving averages of past reparable generation activity. However, at least four quarters of data are always used for these estimates. Hence, to estimate these values at the beginning of FY74-1 we use the D041 data for quarters FY74-1 through FY74-4, since no data prior to FY74-1 is available. To estimate rates to be used in simulating FY77-1, however, we use the D041 data for the eight quarters between FY75-1 and FY76-4. This interval represents the most recent eight-quarters of historical data that would be available at the start of FY77-1.

2. For operating bases, forecasts for future rep gens are based upon historical failure rates and the actual D041 program activity for the future period. Specific LRU installed programs by base are determined by allocating the total LRU program in proportion to the aircraft base programs shown in Figures III-1 and III-2, as appropriate. For forecasts of Aircraft Overhaul requirements, it is assumed that the expected depot rep gen rate may be forecast perfectly over a one year time horizon; however, it is further assumed that errors occur in forecasting the precise time within the year that these depot rep gens occur.

3. All depot reparable generations are assumed to originate from a single aircraft overhaul facility. Stock levels for this facility are computed to provide a 14-day supply.

to a given LRU repairable generation is assumed proportional to the total number of SRU units that are contained in the assemblies of the failed LRUs. For example, for the LRU/SRU pair presented in Table III-1, there are two units of the SRU contained in each LRU; that is, the Quantity Per Application (QPA) for the SRU is two. Consequently, for the 110 LRUs that were repaired in the 110 LRU repairable generations. Further, since there were exactly 10 repairable generations for the SRU during this period, we assume that the probability that any specific SRU component was faulty is 10/220.

In analyzing Air Force historical records, we were unable to relate condmenation actions recorded in one period to specific repairable generation actions recorded in other periods. Consequently, in simulating both depot condemned and depot overhaul condemned actions, we use a probability model which guarantees that the total number of condemnations over the four year simulation period exactly equals the number of condemnations recorded in Air Force repairable generation histories. However, we do not attempt to reproduce the specific quarter-by-quarter condemnation quantities which are recorded in the D041 Depot Data Bank.

Relationships Among Exogenous Events

As noted above, the objective of the Events Generator is to generate detailed descriptions of each LRU and SRU reparable generation in such a way that the totals of these events exactly equal values recorded in D041 reparable generation histories. In doing this, we wish to generate not only events describing specific reparable generations, but also to generate all associated serviceable return, repair, NRTS, and condemnation actions whose timing may be determined from the timing of the reparable generation event. Table III-1 presents a sample of historical reparable generation data which drives the Events Generator, while Table III-2 defines the probability model used to generate specific reparable generation events. On the other hand, Figures III-1 through III-4 describe the relationships between reparable generation events and all associated exogenous events "caused by" the reparable generation.

Figure III-1 illustrates the relationships among exogenous events caused by an LRU base reparable generation. As shown in the figure, an LRU base rep gen event is always accompanied by a corresponding requisition (Event Type 1) for a replacement (serviceable) LRU. In addition, one of three possible additional LRU events may be related to the LRU base rep gen. These are: (a) an LRU base condemnation event (Event Type 15), (b) an LRU RTS event (Event Type 18), or (c) an LRU NRTS event (Event Type 19). Which of these three specific events will be generated depends on a Monte Carlo computation to be discussed later in this section. Suppose that this Monte Carlo calculation indicates that the LRU base reparable generation is to be condemned. As shown in Figure III-1, the next step is

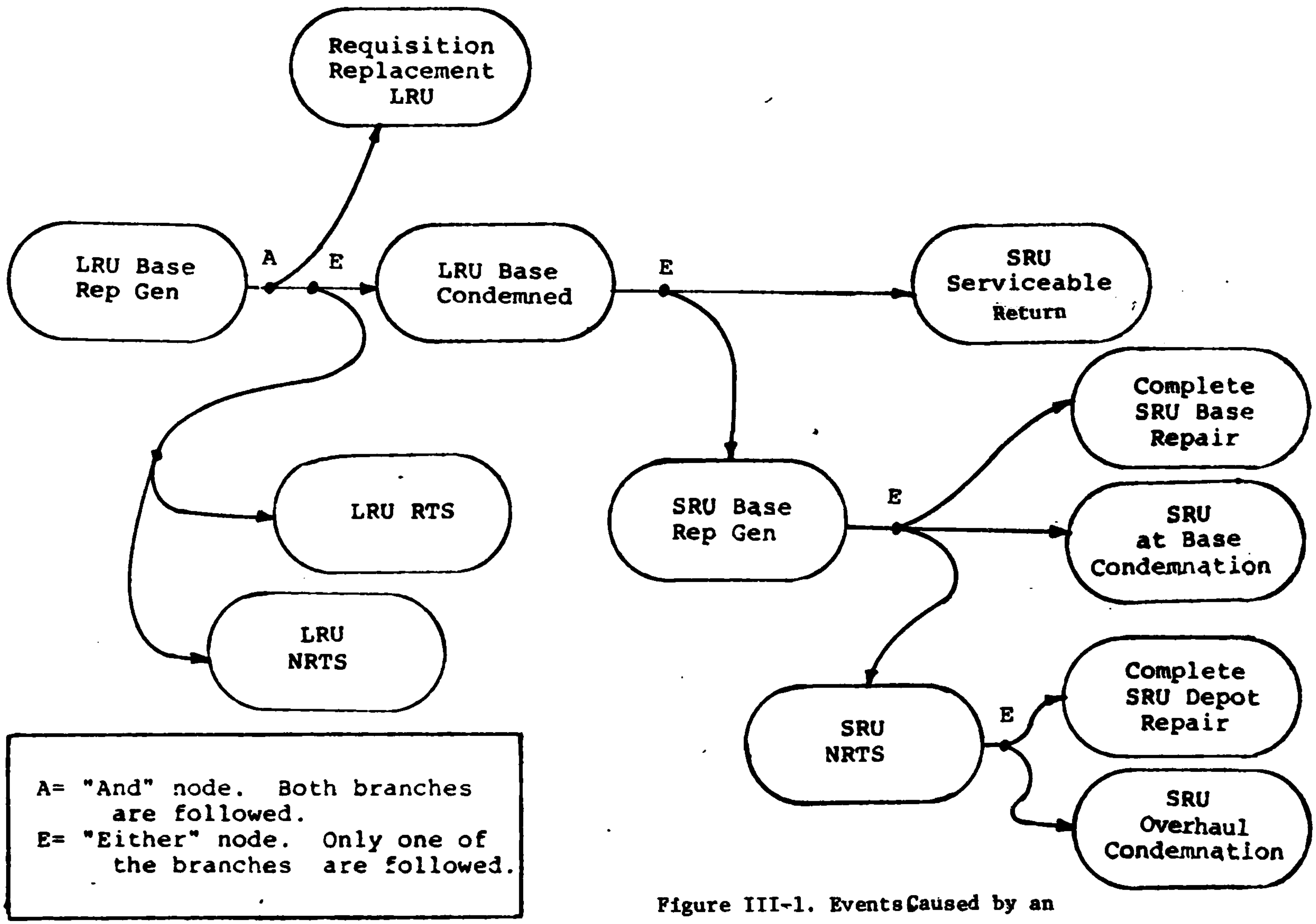


Figure III-1. Events Caused by an LRU Base Reparable Generation.

to determine the relationships between the condemned LRU asset and the individual SRU components which go into the LRU unit. As shown in the Figure, a Monte Carlo process is applied to determine the status of each SRU unit that is a member of the condemned LRU asset. If the Monte Carlo computation indicates that the SRU is serviceable, the Event Generator schedules a Serviceable Return event (Event Type 4) to represent the return of the serviceable SRU to available base stocks. On the other hand, the Monte Carlo computation may indicate that the SRU component is in some way faulty. In this case, the Events Generator schedules a Base Reparable Generation event (Event Type 14) for each faulty SRU component. Monte Carlo techniques are also used to determine the additional exogenous events to associate with the SRU reparable generation. As shown in Figure III-1, one of three *different* exogenous events may be associated with a given SRU Base Reparable Generation event. These are: (a) completion of repair activities for the SRU (Event Type 18), (b) condemnation of the SRU (Event Type 15), or (c) Initiation of a NRTS event (Event Type 19) to represent the beginning of transportation of the faulty SRU to the depot level repair facility. In the latter case, i.e., the case of an SRU NRTS event, a Monte Carlo process is again used to determine the disposition of the SRU when it reaches the Depot Level Maintenance facility. One possibility is that the SRU may be successfully repaired at the depot level. In this case, the Events Generator schedules an SRU repair completion event (Event Type 18) to occur at the depot. The other possibility is that the SRU cannot be repaired. In this case, the Events Generator schedules a Condemnation event (Event Type 15) to occur for the SRU after a suitable time delay to represent the time required to transport the SRU to the depot.

Figure III-2 illustrates the logic for generating exogenous events associated with an LRU rep gen that is repaired at base level. As shown in the figure, one possibility is that the failed LRU may be repaired without the replacement of any of its SRU components. In this case, the Events Generator schedules a repair completion event (Event Type 18) for the LRU to occur after an appropriate repair time delay. On the other hand, repair of the LRU may require the removal and replacement of at least one of its SRU components. In the latter case, the Events Generator first schedules a "Begin Wait" event (Event Type 16) for the LRU, indicating that repair of LRU cannot proceed until suitable SRU replacements are available. As shown in Figure III-2, the Begin Wait event is also accompanied by the generation of exogenous events representing requisitions of all needed SRU components. Once all of the serviceable SRU units required for the repair of the LRU are received, the LRU is removed from the Wait-for-Parts status, and a Repair Completion event for the LRU may be scheduled. These latter activities removing the LRU from its wait status and scheduling repair completion are performed by subroutine RCVPRT within the RIME Simulation Model, and are not part of the Events Generator.

As shown in in Figure III-2, several SRU base reparable generation events may also be scheduled for each faulty SRU contained in the LRU rep gen. In this event, a Monte Carlo process is used to determine if a specific SRU base rep gen is to be (a) repaired at base level (Event Type 18), (b) condemned at the base (Event Type 15), or (c) shipped to the depot for additional repair operations (Event Type 19). In the latter case, Monte Carlo calculations are also used to determine if the SRU is to be repaired at the depot or if it is to be condemned.

EVENTS CAUSED BY AN LRU BASE REPARABLE GENERATION

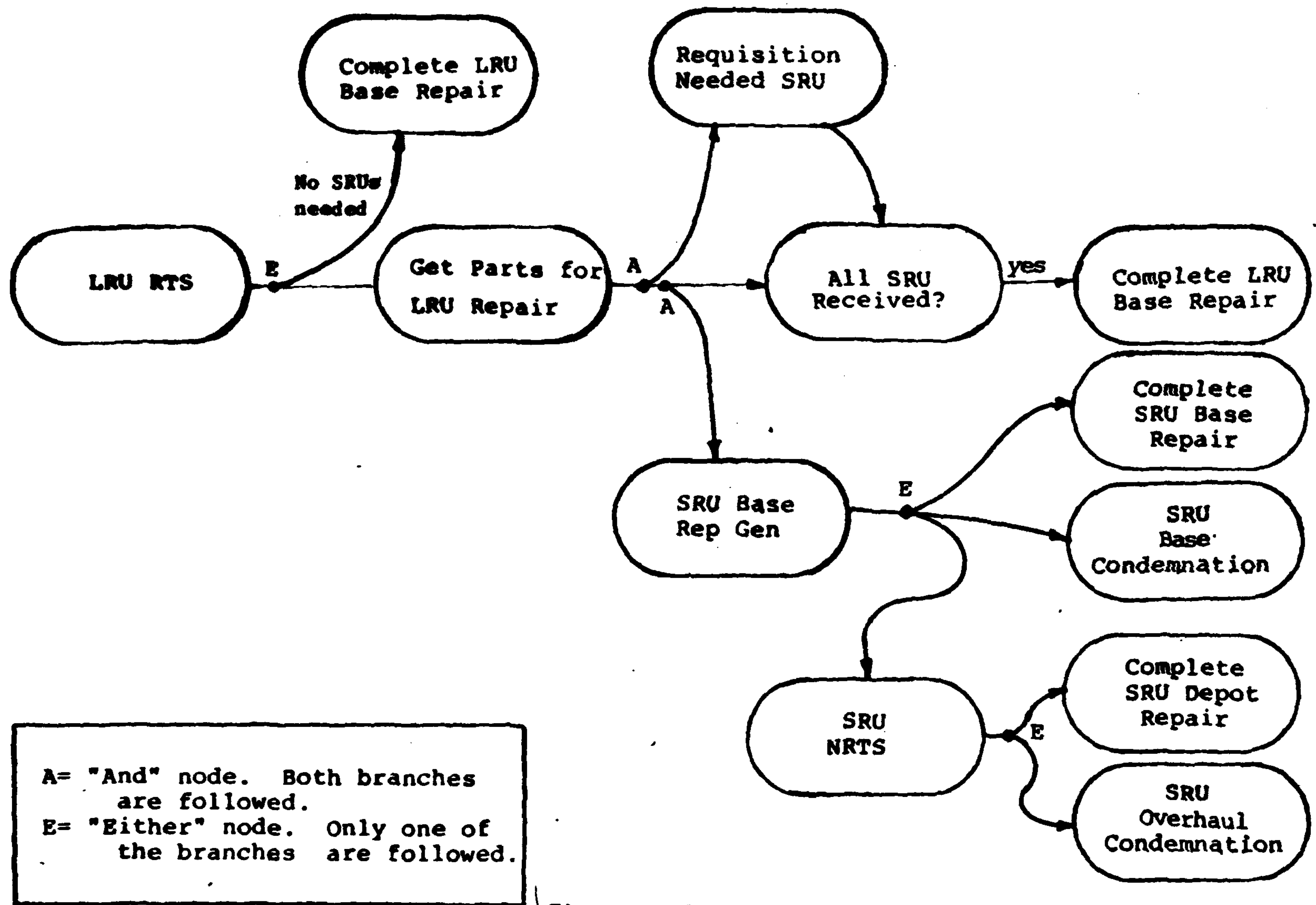


Figure III-2. Events Caused by an LRU RTS Event.

As shown in Table III-2, we attempt to relate all SRU base reparable generations to LRU reparable generations. However, this is not always possible. At times, D041 histories show more SRU base reparable generations then can be explained by the failure of every SRU installed in all recorded LRU failures. In RIME, we assume that all SRU reparable generations which cannot be associated with LRU failures are "independent" base reparable generations. Such generations might occur, for example, if faulty SRUs are removed from an LRU at the flightline, rather than at an intermediate repair organization. In this case, the LRU reparable generation would never be recorded in the D041 demand history, while the SRU reparable generation would be recorded.

Figure III-3 illustrates the relationships among all exogenous events associated with Independent SRU base reparable generation events. As may be seen in the figure, these are essentially the same event relationships as when the SRU is associated wtih an LRU failure.

Figure III-4 illustrates the relationship among exogenous events associated with the Depot Level Maintenance facility. As shown in the figure, there is no attempt to relate LRU repair or condemnation actions at the depot level with SRU repair or condemnation events. Basically, this flowchart assumes the same LRU/SRU depot relationships as those modeled in the METRIC and MOD-METRIC mathematical models.

Figure III-3.

Events Caused by An Independent SRU Base Reparable Generation.

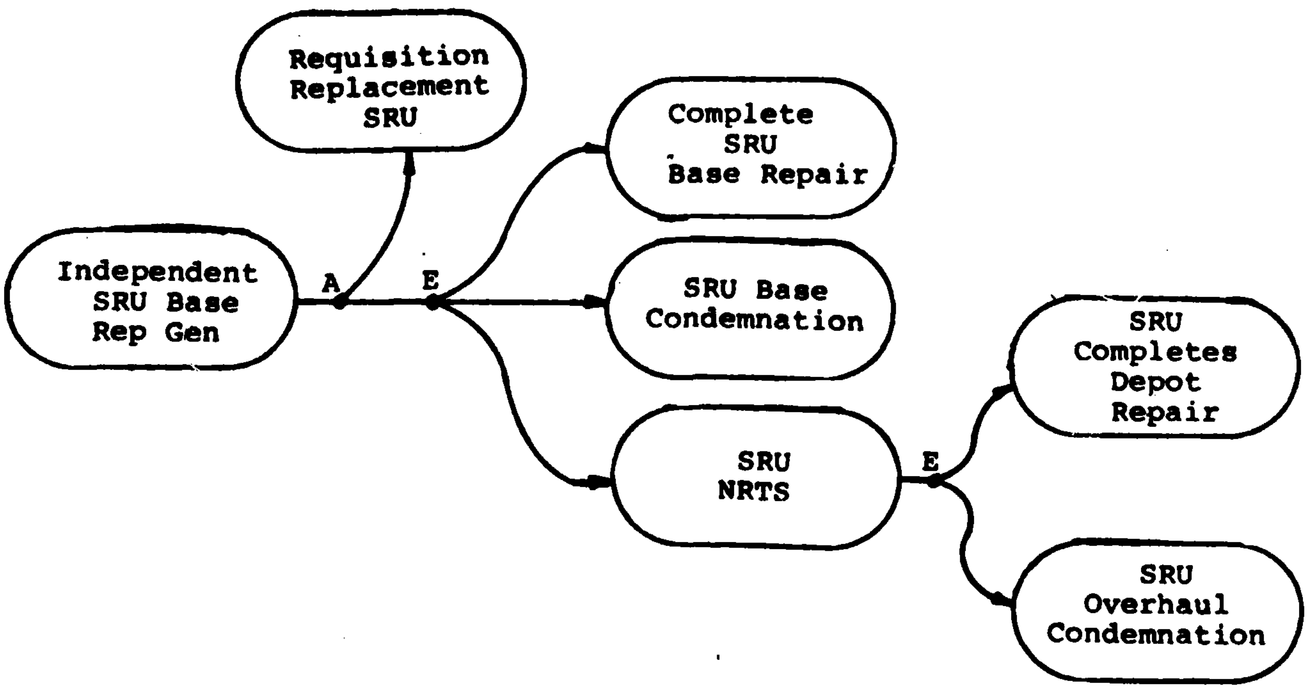
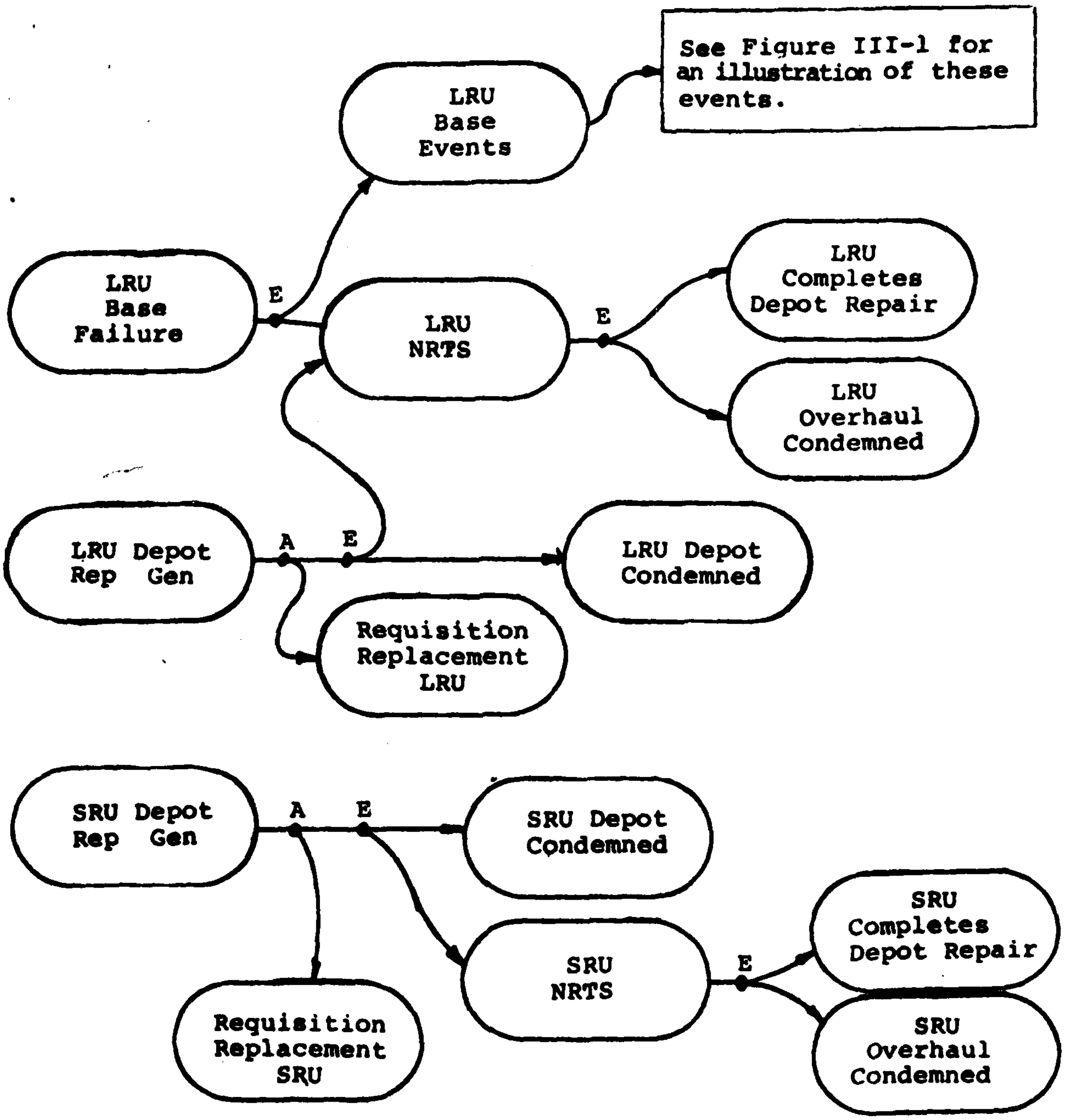


Figure III-4.
Depot Level Events



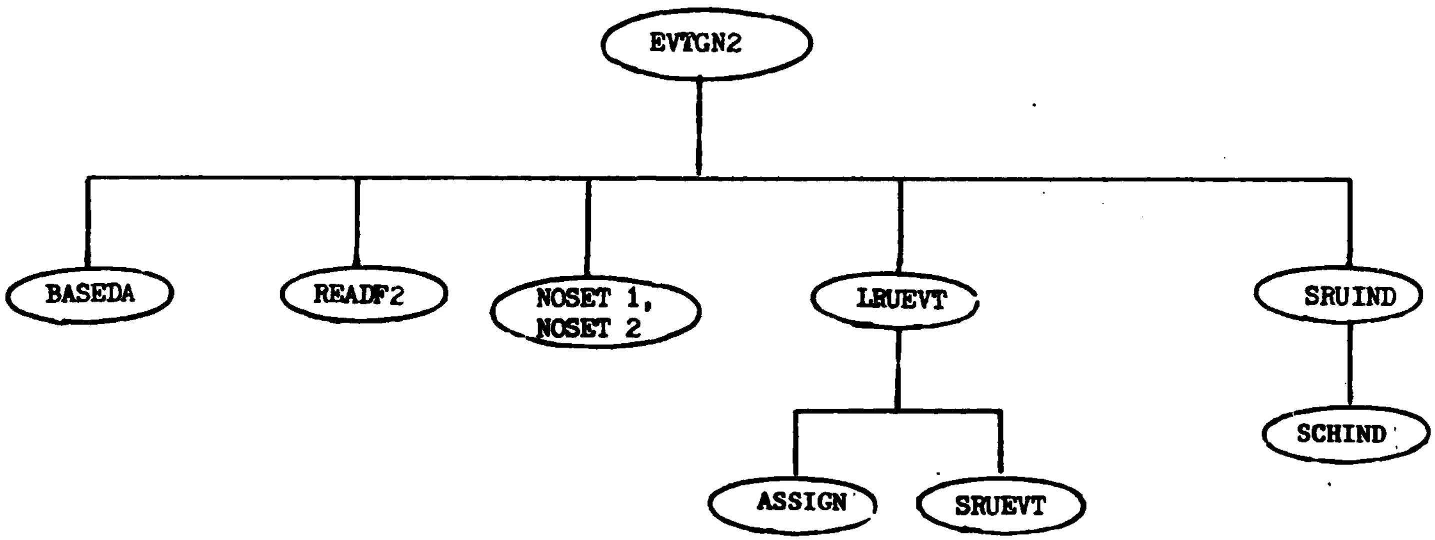
Note: Job-Routed Depot Reparable Generations are not recorded in D041, and are thus not simulated in RIME.

Programs Used in the Events Generator

Figure III-5 illustrates the computer programs which implement the Events Generator. As shown in the figure, the program EVTGN2 is the main program for the events generation process. The subroutines BASEDA and READF2 are used to input base flying program data by quarter and LRU/SRU item data, respectively. The subroutines NOSET1 and NOSET2 are used to set counters required to control the events generation process. Subroutine NOSET1 sets counters of depot level events, while NOSET2 sets base level counters. The routines LRUEVT and SRUEVT perform the bulk of calculations in the Events Generator. The routine LRUEVT controls the scheduling of LRU reparable generations and all associated LRU exogenous events, and LRUEVT calls subroutine SRUEVT to generate all related SRU events. In some cases, D041 histories show more SRU reparable generations than can be explained by the historical values for LRU reparable generations. In this case, the subroutine SRUIND is called to schedule all "excess" SRU reparable generations as "independent" events. In turn, SRUIND calls subroutine SCHIND to perform the detailed scheduling calculations associated with these independent events.

In addition, three general utility routines -- INFEL, ENTER, and REMOVE -- are called to facilitate the scheduling of exogenous events. Subroutine INFEL is called to initialize the Future Events List (FEL), while subroutine ENTER is called to place specific events on the FEL. At the end of each quarter, subroutine REMOVE is called to remove all events associated with the given quarter from the Future Events List. Let us now consider interrelationships among these routines in more detail.

Figure III-5.
Program Relationships in the Events Generator



UTILITY ROUTINES

INFEL
ENTER
REMOVE

EVTGN2: The Exogenous Event Generator Main Program

The program EVTGN2 is the main program for the Exogenous Event Generator. A flow chart of the major computations performed by EVTGN2 is shown in Figure III-6, while detailed source program listings of EVTGN2 are presented in Volume II. As shown in the figure, EVTGN2 begins by reading parameters which control the computations performed by the Events Generator, and which control the types of outputs to be produced during the Events Generation Process. The run parameters read in are:

NFGRP = The number of the first LRU/SRU group included in the events generation process.

NLGRP = The number of the last LRU/SRU group be included in the events generation process.

NBASES = The number of operating bases assumed during events generation process.

INQTR = The number of quarters in the events generation planning horizon.

NREPL = The number of replications to be performed for each LRU/SRU group.

IWT(I) = A "Write" Flag, which specifies if output option I is to be exercised. If IWT(I) = 1, the Ith print option is to be used; otherwise, that option is not used.

The Write Flags are most useful during the debugging of new data sets and in the development of alternate versions of the Events Generation System. Table III-3 defines the specific meanings of each of these Flags.

After reading the run parameters and Write Flags, subroutine BASEDA is called to read in base order and ship times. BASEDA also reads in flying program data by base for each quarter in the events generation planning horizon. Next, EVTGN2 scans down the Master Data Set until LRU/SRU group NFGRP is found. The routine then begins the generation of exogenous events with this particular group.

In processing each new LRU/SRU group, the replication counter KREPL is reset to 1, and the reparable generation counter NJOB is set to 1,000. The Future Events List is then initialized by calling subroutine INFEL, and the counters of depot level events are initialized by a call to subroutine NOSET1.

The quarter loop now begins by setting the quarter counter variable IQTR = 1. At the beginning of each quarter, subroutine RANBS1 is called to initialize the base probability array CPROB, and subroutine NOSET2 is called to initialize the counters of all base level events to be generated within the current quarter. Probability array CPROB is used by subroutine RANBAS later in the events generation process to determine a randomly selected base. This selection is performed so that the base selection probability is proportional to the flying hour activity at each base in specific quarter under consideration.

Figure III-6.

The Exogenous Event Generator

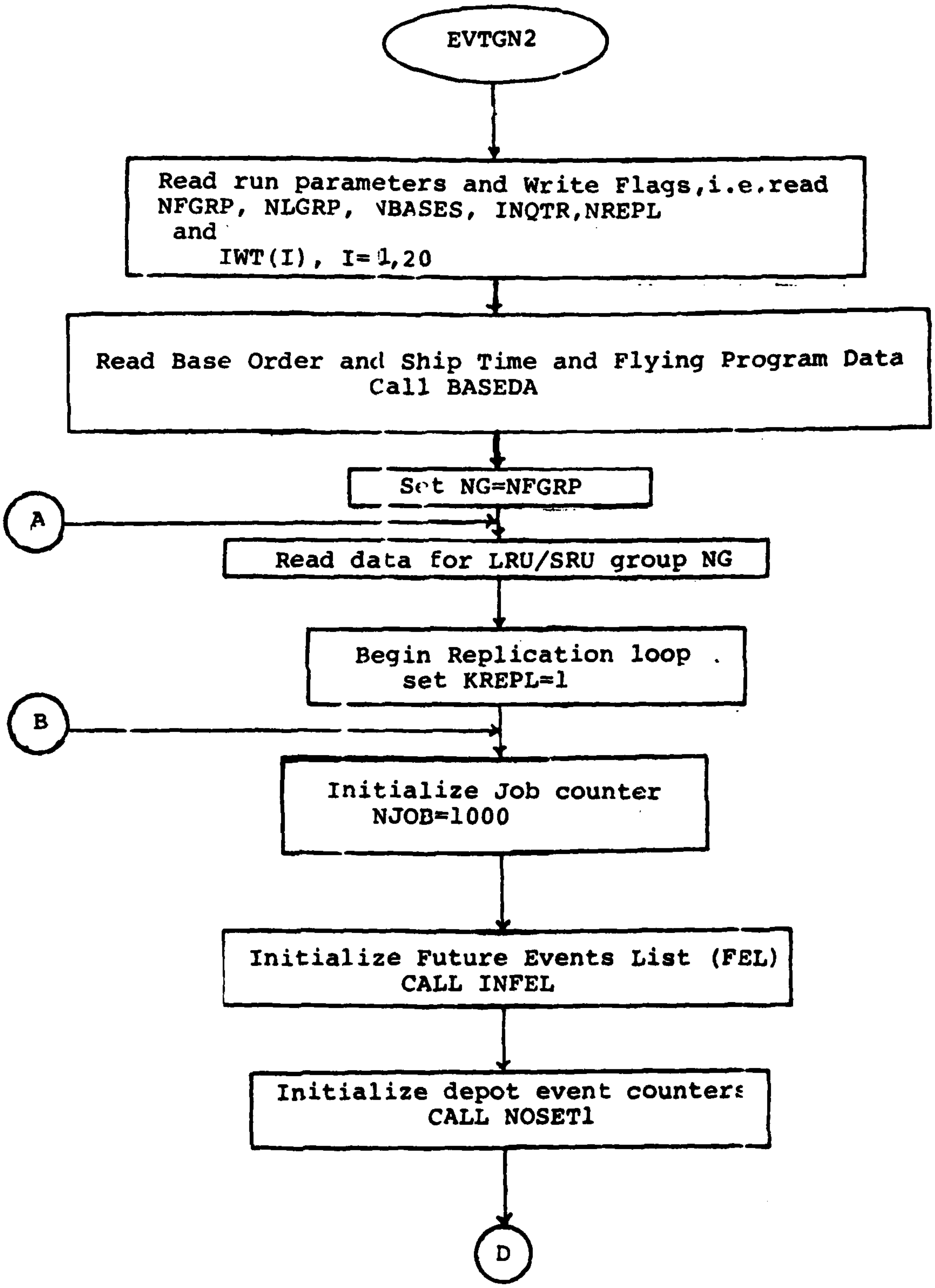


Figure III-6 (Continued)

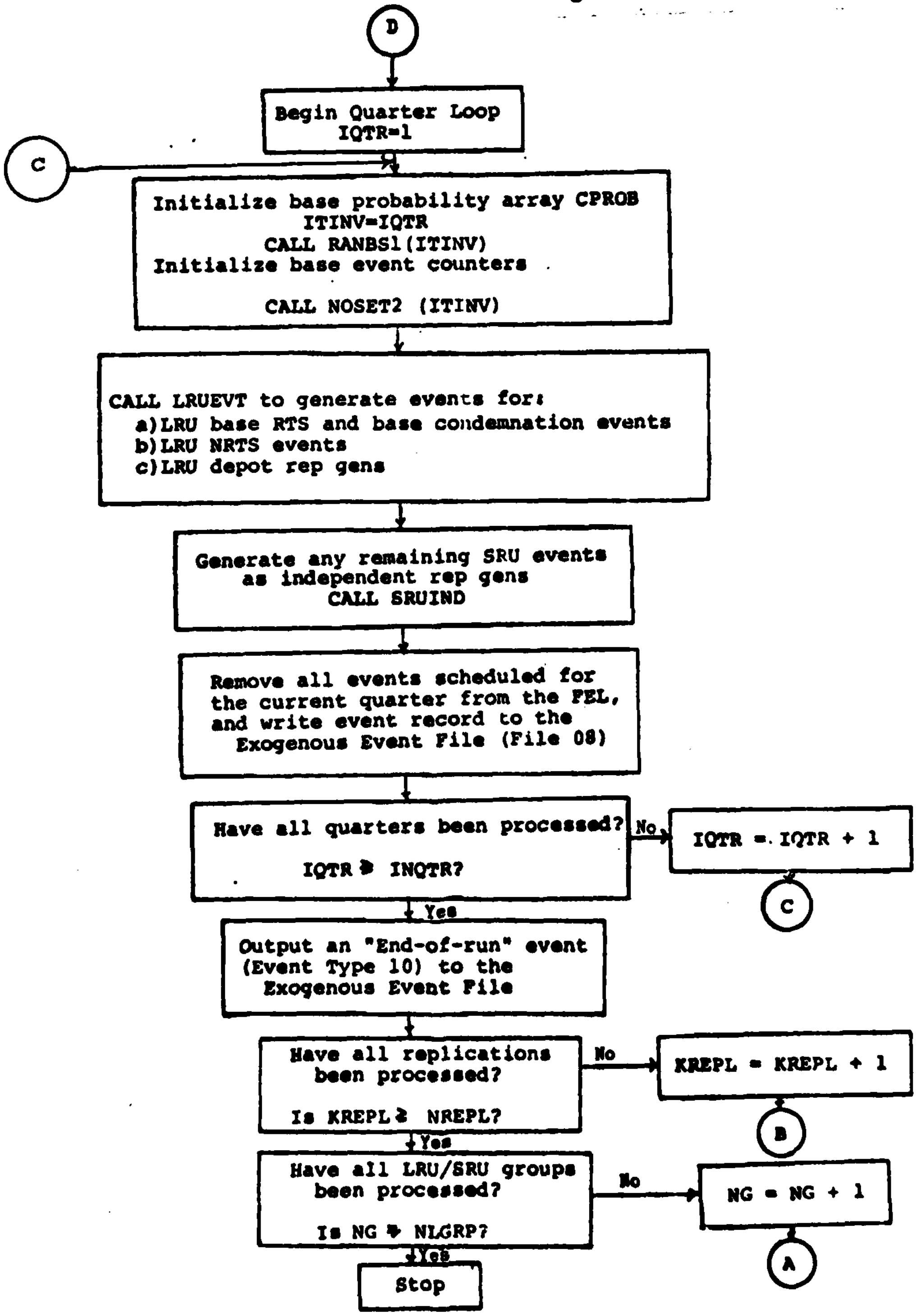


TABLE III-3

EVENTS GENERATOR

WRITE FLAGS

Write Flag Number	Routine	Effect
1	READFL	Print all data read in, where (1) = print short header only, (2) = print header and rep gen data, (3) = print all data input
2	READFL	Print header data written to EXOGFILE
3	NOSET1 NOSET2	Print values for all event counters
4	LRUEV2	Print number of failed SRU components in each LRU rep gen
5	ASSIGN	Print details of individual component Monte Carlo failure calculations
6	SRUEVT	Print details of dependent SRU events generated by SRUEVT
7	SCHIND	Print details of independent SRU event generation
8	EVNTGN2	Print values of LRU counters
9	EVNTGN2	Print events written to File 08
10	ENTER, REMOVE	Print all entries and removals to the Future Events List
11	EVTGN2	Set IOUT = 1, and produce a binary exogenous event file on File 08.
12	BASEDA	Write base data
13	RANBS1	Write base probabilities

After the above initialization procedures, subroutine LRUEVT is called three times to generate all exogenous events associated with historical D041 LRU reparable generations associated with quarter IQTR. First, LRUEVT is called to generate specific events associated with historical LRU base RTS and base condemnation events. The second and third calls to subroutine LRUEVT generate all exogenous events associated with LRU NRTS and LRU depot reparable generations, respectively. In performing the LRU event generation process, subroutine ASSIGN is called to determine the number of SRU reparable generations to associate with each LRU failure, while subroutine SRUEVT is called to schedule the specific exogenous events associated with these SRU reparable generations.

After all LRU events have been scheduled, subroutine SRUIND is called. This routine checks the base level counter arrays for each SRU to determine if there are any remaining SRU reparable generations, i.e., SRU generations which could not logically be associated with LRU failures. If so, subroutine SCHIND is called to schedule the remaining SRU events as "independent" SRU reparable generations, and to schedule all related requisition, cancellation, and repair activities.

Finally, at the end of each simulated quarter, subroutine REMOVE is called to remove all exogenous events scheduled for quarter IQTR from the FEL, and to output an event record to the Exogenous Event File (File 08). The quarter counter IQTR is then increased by one, and the event generation process continues until all quarters have been considered.

After all quarters have been processed, an "End-of-Run" (Event Type 10) event record is written to the Exogenous Event File. The RIME Simulation Model uses the "End-of-Run" event record to determine that there are no additional events for the current replication of a given LRU/SRU group. If additional

replications are required, program control returns to the beginning of program EVTGN2, and a new set of events are generated using different values for random number seeds. The event generation process is then repeated using different random values until all replications have been performed. After all required replications are completed, EVTGN2 reads the next LRU/SRU data set from the Master Data File Set, and the events generation process is performed for the new LRU/SRU group. This process continues until all LRU/SRU groups from NFGRP through NLGRP have been processed.

Variable Definitions and Relationships

A large number of variables are required to keep track of the number of reparable generation, RTS, NRTS, and condemnation events remaining to be generated for each LRU and SRU. The major variables used to do this are shown in Table III-4. The depot event counters are initialized by subroutine NOSET1 at the beginning of quarter 1 for each replicatiuon, while counters of base-level events are reset each quarter by subroutine NOSET2. Each of these variables are then decremented as the associated events are scheduled. Table III-5 presents the calculation formulas used to determine the probabilities of specific exogenous events used in the Monte Carlo calculations, while Figure III-7 provides a graphical description of the relationship among these variables. In general, variables starting with an "L" denote counts for the LRUs, while variables beginning with a "N" denote counts for SRUs. Variables starting with a "I" denote variables which are provided as input to the Events Generator through subroutine READFL. Definitions of these latter input variables may be found in the record layouts section of Volume II. Finally, variables ending in "T" denote the total number of assets summed over all periods in the planning horizon.

TABLE III-4

EVENTS GENERATOR
VARIABLE DEFINITIONS

VARIABLE	DEFINITION
NLRUGP	Number of LRU/SRU groups to be simulated.
NBASES	Number of Operating Bases.
NSRU	Number of SRUs in the LRU.
NITEM	Number of stock-keeping units (SKUs).
NMPH	Number of months of program data provided.
INQTR	Number of quarters of events to be generated.
FH(I,J)	Flying program (in 100s of hours) in quarter I at base J.
TFH(J)	Total flying program at base J for months 1, 2,....,NMPH.
ITINV	Current Quarter index.
ITDAY	Time units in 1 day (ITDAY = 100 Time Units).
ITQTR	Time units in 1 quarter = 8400 (7 days/week x 4 weeks/month x 3 months/quarter).

<u>For a given quarter, let K denote the Kth type of SRU in an LRU. Then,</u>

NOINL(K)	Number of SRU Type K failures in the LRU reparable generation currently being considered.
NOSRUF(K)	Total number of base level RTS, NRTS, and condemnation events remaining to be assigned to rep gen events.
NORTS(K)	Number of base level Reparable This Station (RTS) SRU Type K generations remaining to be assigned to rep gen events.
NONRTS(K)	Number of base level NRTS generations remaining to be assigned to rep gen events.
NOCON(K)	Number of base condemnations remaining to be assigned to rep gen events.

TABLE III-4 (CONT'D)

NODCN(K)	Number of depot condemnations remaining to be assigned to rep gen events.
NODRP(K)	Number of depot repairs remaining to be assigned to rep gen events.
LBRGN	Number of base level LRU failures in the current quarter (total LRUs recorded as RTS or condemned.
NOLEFT	Number of LRU failures that have not yet been scheduled.
LRTS	Number of LRU RTS events that have not yet been scheduled.
NJOB	Job number associated with the current rep gen and to all related exogenous repair events.
LNRTS	Number of LRU NRTS events in the current quarter.
LDRGN	Number of LRU depot rep gens in the current quarter.
LDCON	Number of LRU depot condemnations in the current quarter.
LDCONT	NUMBER OF LRU depot condemnations over the planning horizon.
LOVGNT	Number of NRTS and depot LRU generations less depot condemantions over the planning horizon. LOVGNT thus represents the total number of LRU carcasses received at the depot over the planning horizon.
LDREP	Number of the LOVGNT LRUs that are repaired.
LOVCNT	Number of depot LRU overhaul condemnations over the planning horizon.
NDRGN(K)	Number of depot generations of SRU K over the planning horizon.
NINDGN (I,K)	Number of independent depot generations for SRU K in quarter I.
NDCOND(K)	Number of depot condemnations of SRU K.
NOVCNT(K)	Number of overhaul condemantions of SRU K over the planning horizon.
NSRUGN(K)	Total SRU NRTS plus total SRU depot rep gens over the planning horizon.
LRUREP =	LRU Repair Status Flag. 0 indicates the LRU is to be condemned. 1 indicates the LRU is repairable.
LRULOC =	Base number associated with a given LRU condemnation, where LRULOC = 0 indicates a depot condemnation.

TABLE III-5

MONTE CARLO CALCULATION FORMULAS

A. LRU Base Failure Probabilities

$$P\left[\begin{array}{l}\text{LRU is}\\ \text{Condemned}\end{array}\right] = \text{LBCOND/LBRGN}$$

$$P\left[\begin{array}{l}\text{LRU is}\\ \text{RTS}\end{array}\right] = \text{LRTS/LBRGN}$$

$$P\left[\begin{array}{l}\text{SRU K is defective}\\ \text{in a given LRU base}\\ \text{rep gen}\end{array}\right] = \text{NOSRUF(K)/ LBRGN*IQPA(K)}$$

B. Given that a particular SRU K unit is defective,

$$P\left[\begin{array}{l}\text{SRU K}\\ \text{is RTS}\end{array}\right] = \text{NBRTS(K)/(NBRTS(K) + NBCOND(K) + NNRTS(K))}$$

$$P\left[\begin{array}{l}\text{SRU K}\\ \text{is condemned}\end{array}\right] = \text{NBCOND(K)/(NBRTS(K) + NBCOND(K) + NNRTS(K))}$$

$$P\left[\begin{array}{l}\text{SRU K}\\ \text{is NRTS}\end{array}\right] = \text{NNRTS(K)/(NBRTS(K) + NBCOND(K) + NNRTS(K))}$$

C. The number of independent base SRU rep gens, NOIND(K), is computed as follows:

If LBRGN*IQPA(K) NOSRUF(K), then NOIND(K) = 0
Otherwise, NOIND(K) = NOSRUF(K) - LBRGN*IQPA(K)

D. SRU depot repair probabilities

$$P\left[\begin{array}{l}\text{SRU NRTS}\\ \text{is}\\ \text{Overhaul Condemned}\end{array}\right] = \text{NOVCNT(K)/(NOVCNT(K) + NODRPT(K))}$$

$$P\left[\begin{array}{l}\text{SRU NRTS}\\ \text{is}\\ \text{Repaired}\end{array}\right] = \text{NODRPT(K)/(NOVCNT(K) + NODRPT(K))}$$

Figure III-7

Relationships Among Events Generator Variables

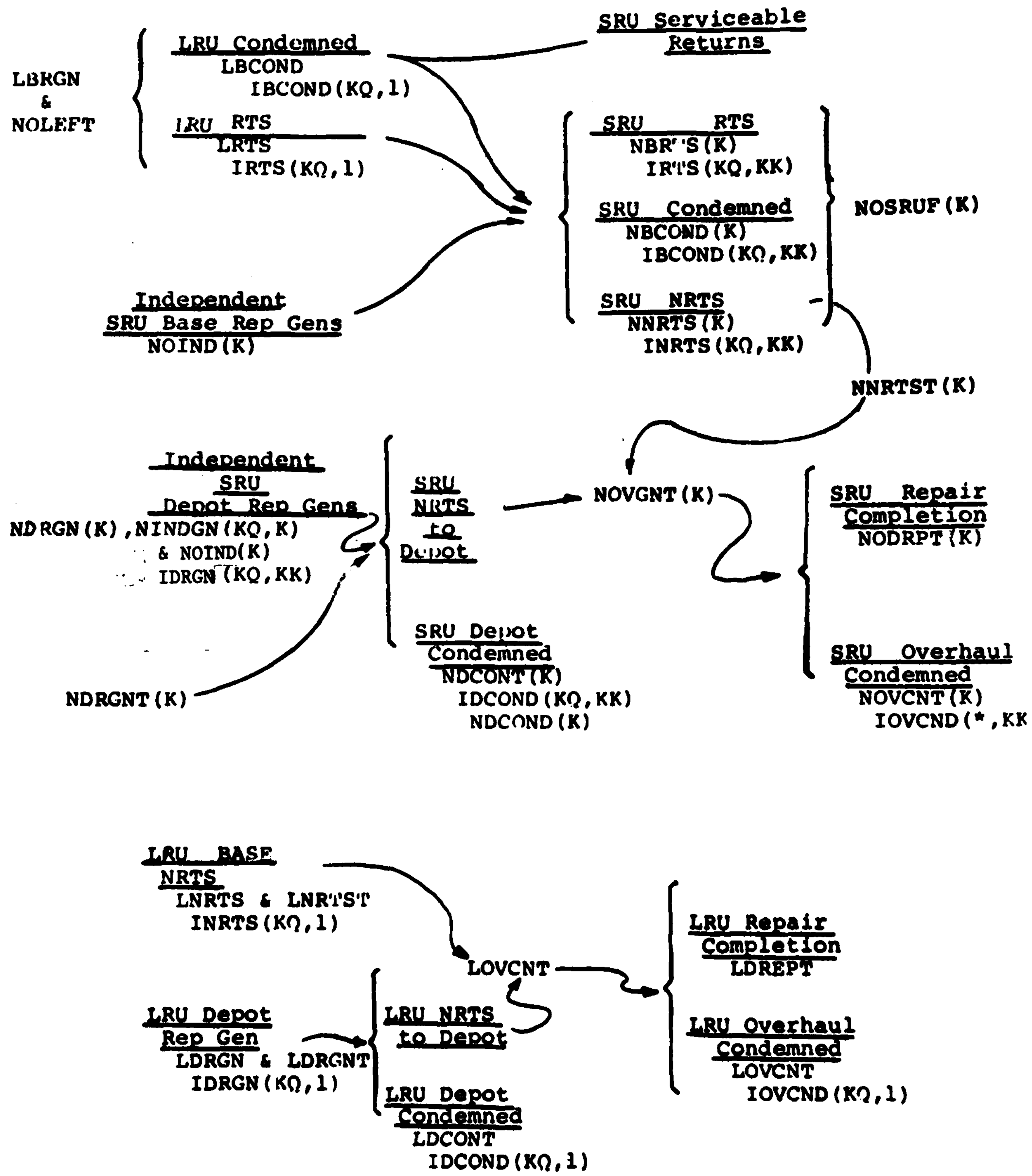


FIGURE III-7 (CONTINUED)

DESCRIPTION OF SYMBOLS

SYMBOL	DEFINITION
*	Values are constructed to be exactly the same as historical values on a quarter-by-quarter basis.
+	Values are constructed so that the totals over the planning horizon exactly equal historical values.
	These values are derived from other variables.
/	These values are limited to be no more than the total asset flow through this point.
K	SRU Number.
KK	K + 1.
KQ	Quarter number.

An Example

If the Write Flag IWT(10) = 1, all events which are entered or removed from the Future Events List are printed. For example, Figure III-8 illustrates a set of 34 events placed on the FEL during a test run of the Events Generator. As shown in the figure, these events are related to three individual LRU reparable generations. These reparable generations are assigned job numbers of 1001, 1002, and 1003, respectively. The first LRU reparable generation event (Event Code 14) is scheduled to occur at base #3 at time 4444. Consequently, the Stock Keeping Unit number associated with this event is 4. (The reader should review the SKU numbering rules discussed in Chapter II to verify this). Figure III-8 shows the time, event code, Stock Keeping Unit number, quantity, and "priority/job number" code associated with each generated event. Consequently, the first line of this Figure shows that a Code 14 event is scheduled at time 4444 for SKU = 4. Further, this line shows that only 1 asset is involved and that a reparable generation number NJOB = 1001 has been assigned to this reparable generation event. The second line of Figure III-8 shows that a requisition to replace the failed LRU is scheduled to occur at time 4454. This requisition has an Event Code = 1, the SKU = 4, a quantity of 1 unit is involved, and a priority code of 1 has been assigned indicating this is a high priority requisition.

The third line of Figure III-7 shows that at time 4449 a type 16 event is scheduled, indicating that the LRU(SKU = 4) begins to wait for SRU components associated with the LRU failure at that time. For this event, the quantity field equals 3, indicating that the total number of SRU components required to complete

FIGURE III-8

SAMPLE EVENT GENERATION

Event Family	Events Generation Sequence		Event Time	Event Code	SKU	QTY	Priority and Job No. Code
1001	1	LRU Rep Gen, Base 3	4444	14	4	1	1001
	2	Req. Replacement LRU	4454	1	4	1	1
	3	WAIT for SRU's (3)	4449	18	4	3	1001
	4	SRU#1 Rep Gen, Base 3	4459	14	9	1	1001
	5	Req. Replacement SRU	4464	1	9	1	100102
	6	Complete SRU Repair	5444	18	9	1	1001
	7	SRU#2 REP GEN, BASE 3	4459	14	14	2	1001
	8	REQ. REPLACEMENT	4464	1	14	2	100102
	9	CONDEM FIRST UNIT	4464	19	14	1	1001
	10	CONDEM SECOND UNIT	4464	19	14	1	1001
1002	11	LRU REP GEN AT BASE #2	2643	14	3	1	1002
	12	REQ. REPLACEMENT	2653	1	3	1	1
	13	WAIT FOR SRUs (3)	2648	18	3	3	1002
	14	SRU#1 Rep Gen at Base #2	2658	14	8	1	1002
	15	REQ REPLACEMENT SRU	2663	1	8	1	100202
	16	CONDEM THE REP GEN	2663	15	8	1	1002
	17	SRU#2 REP GEN AT BASE #2	2658	14	13	2	1002
	18	REQ. REPLACEMENTS	2663	1	13	2	100202
	19	NRTS FIRST REP GEN	2673	19	13	1	1002
	20	COMPLETE DEPOT REPAIR	7643	18	11	1	1002
	21	NRTS 2ND REP GEN	2673	19	13	1	1002
	22	CONDEM 2ND " " AT DEPOT	4643	15	11	1	1002
1003	23	LRU REP GEN, BASE #3	498	14	4	1	1003
	24	REQ. RELACEMENT LRU	508	1	4	1	1
	25	WAIT FOR SRUs (3)	503	18	4	3	1003
	26	SRU#1 REP GEN, BASE #3	513	14	9	1	1003
	27	REQ. REPLACEMENT SRU	518	1	9	1	100302
	28	NRTS REP GEN TO DEPOT	526	19	9	1	1003
	29	COMPLETE REPAIR AT DEPOT	5498	18	6	1	1003
	30	SRU#2 REP GEN, BASE #3	513	14	14	2	1003
	31	REQ. REPLACEMENTS	516	1	14	2	100302
	32	NRTS 1ST REP GEN	528	19	14	1	1003
	33	REPAIR 1ST " " AT DEPOT	5498	18	11	1	1003
	34	REPAIR 2ND REP GEN AT BASE	1498	18	14	1	1003

NJOB = 1001

NNEED = 3

Two REP GENS

NNEED = 3

repair of the LRU is NNEED = 3. Since this Begin Wait event is associated with LRU reparable generation 1001, the job number 1001 is shown in the right hand column for this event.

Lines 4 through 10 of Figure III-8 show the SRU events associated with LRU reparable generation number 1001. Events 4 and 5 represent the generation of one failed unit of SRU #1 at base 3, and an associated requisition for a serviceable unit, respectively, while event 6 indicates that repair of this failed SRU will be completed at time 5444. On the other hand, events 7 through 10 describe exogenous events associated with reparable generations of SRU #2. Event 7 records the generation of 2 failed units of SRU #2; the right hand column indicates these failures were discovered during the repair of LRU job number 1001. Event 8 indicates a requisition for 2 serviceable units to replace the failed components will reach base supply at time 4464. On the other hand, events 9 and 10 describe the disposition of the failed assets of SRU #2. Event 9 indicates that the first unit of SRU #2 to be removed from LRU reparable generation 1001 is to be condemned (Event Code 15) at time 4464, and event 10 shows the second unit of SRU #2 removed from LRU job number 1001 is also condemned at base level. This second condemnation is also scheduled to occur at time 4464. This completes the list of exogenous events associated with LRU reparable generation number 1001.

Events 11 through 22 describe the events associated with the second LRU reparable generation. This second reparable generation is assigned the job number 1002. As shown for event 11, this LRU reparable generation is scheduled to occur at base number 2 at time 2643, while event 12 indicates that the requisition for a serviceable replacement is scheduled to reach base supply at time 2653. In this case, three replacement SRU units are needed to return LRU to 1002 a serviceable

condition. Event 13 records the start of a "Begin Wait" status for the LRU. The LRU will remain in this status until all three of the required SRU components are obtained from base supply. Events 14 through 22 record the generation of the three faulty SRU units associated with LRU failure 1002, and the subsequent disposition of these units. Events 14, 15, and 16 record the failure of a single unit of SRU #1 at base #2 and its subsequent condemnation at that base (Event Number 16) at time 2663. On the other hand, events 17 through 22 describe the generation and subsequent disposition of two faulty units of SRU #2 associated with LRU rep gen 1002. Event 17 records the generation of 2 faulty units of SRU #2 at time 2658, while event 18 indicates that requisitions for serviceable replacements of these units reach base supply at time 2663. Event 19 indicates that the first of these failed SRU #2 units is shipped to the depot (a NRTS event) at time 2673, while event 20 indicates that this NRTS asset completes depot repair at time 7643. Finally, event 21 shows that the second failed unit of SRU #2 associated with LRU rep gen 1002 is also sent to the depot for repair. However, event 22 indicates that this second asset is condemned at the depot at time 4643.

Events 23 through 34 describe the exogenous events associated with the third LRU reparable generation, job number 1003. In this case, the LRU reparable generation is scheduled to occur at time 498. Events associated with job 1003 follow a pattern similar to those described above and will not be discussed further here.

The specific timing for a given LRU reparable generation is determined by selecting a random number which is uniformly distributed over the quarter of interest. Once the specific timing of LRU rep gen has been determined, all related exogenous events are scheduled by adding appropriate time delays to the time of

LRU reparable generation. Individual exogenous events are scheduled (i.e., put on the FEL) as soon as the need for the event is established, regardless of the specific timing of the event. This explains why the first LRU reparable generation (job 1001) -- which is scheduled to occur at time 4444 -- is placed on the FEL before LRU rep gen 1003, which is scheduled to generate at time 498. For output, however, all exogenous events must be ordered by ascending event times. This sorting of events into time sequence is performed by subroutines ENTER and REMOVE. When the need for a given event is determined, subroutine ENTER is called, and the five data elements for the event is placed into its appropriate time sequence on the Future Events List. At the end of each quarter, subroutine REMOVE is called to remove all of those events scheduled to occur within the given quarter.

If the Write Flag IWT(10) = 1, all events that are removed from the Future Events List are printed. Figure III-9 shows the list of events removed from the Future Events List subsequent to the scheduling activities described in Figure III-8. As shown in Figure III-9, the first event to be removed from the FEL is the Event Type 14 event associated with reparable generation 1003. This event is scheduled to occur at time 498. Although this was the 23rd event to be generated (see Figure III-8), it has the lowest scheduled clock time, and consequently is the first event to be removed from the Future Events List. Similarly, the second event to be removed from the FEL is the Begin Wait event (Event Type 16) associated with LRU reparable generation 1003. The reader may verify that the other elements shown in Figure III-9 correspond to the time-sorted values of Figure III-8.

Each time an event is removed from the Future Events List, a corresponding record is written to the Exogenous Event File (File 08). Consequently, at the end of an Events Generator run, File 08 contains a list of exogenous events in ascending time sequence.

FIGURE III-9
EVENTS GENERATOR OUTPUT

	Event No.	Time	Event Code	SKU	QTY	Priority and Job No.	Generation Sequence
REMOVE	1	498	14	4	1	1003	23
REMOVE	2	503	16	4	3	1003	25
REMOVE	3	508	1	4	1	1	24
REMOVE	4	513	14	9	1	1003	26
REMOVE	5	513	14	14	2	1003	30
REMOVE	6	518	1	9	1	100302	27
REMOVE	7	518	1	14	2	100302	31
REMOVE	8	528	19	9	1	1003	28
REMOVE	9	528	19	14	1	1003	32
REMOVE	10	1498	18	14	1	1003	34
REMOVE	11	2643	16	3	1	1002	11
REMOVE	12	2648	16	3	3	1002	13
REMOVE	13	2653	1	3	1	1	12
REMOVE	14	2658	14	8	1	1002	14
REMOVE	15	2658	14	13	2	1002	17
REMOVE	16	2663	1	8	1	100202	15
REMOVE	17	2663	15	8	1	1002	16
REMOVE	18	2663	1	13	2	100202	18
REMOVE	19	2673	19	13	1	1002	19
REMOVE	20	2673	19	13	1	1002	21
REMOVE	21	4444	14	4	1	1001	1
REMOVE	22	4449	16	4	3	1001	3
REMOVE	23	4454	1	4	1	1	2
REMOVE	24	4459	14	9	1	1001	4
REMOVE	25	4459	14	14	2	1001	7
REMOVE	26	4464	1	9	1	100102	5
REMOVE	27	4464	1	14	2	100102	8
REMOVE	28	4464	15	14	1	1001	9
REMOVE	29	4464	15	14	1	1001	10
REMOVE	30	4643	15	11	1	1002	22
REMOVE	31	5444	18	9	1	1001	6
REMOVE	32	5498	18	6	1	1003	29
REMOVE	33	5498	18	11	1	1003	33
REMOVE	34	7643	18	11	1	1002	20

CHAPTER IV

The Levels Computation System

The Levels Computation System provides the capability to compute LRU and SRU stock levels using the METRIC, MOD-METRIC, VSL, and AFLCR 57-27 of requirements computation methodologies, as well as variations of these methods. This Chapter describes the major features of this system.

Stock Level Computations

The management of recoverable item inventories involves providing answers to three basic questions. These are:

1. When will recoverable spares be needed?
2. How much is needed?
3. Where should the spares be located?

The first two questions are Procurement issues, for they determine how much stock is needed to support a given system, and when this stock should be acquired and brought into the Air Force supply system. The third question, on the other hand, is a Distribution issue. The answer to this question determines at which locations the currently available stock should be positioned.

Two distinct phases may be identified during the life cycle of a given item. These are the Initial Provisioning Phase and the Replenishment Phase. The Initial Provisioning Phase determines the number of assets which will be acquired during the initial buy of an asset, while the Replenishment Phase determines the number of additional assets which are required to compensate for (a) condemnations

resulting from operations, (b) unexpectedly high failure rates, or (c) increased levels of program activity. Different computational methods may be used in each of these phases to determine recoverable item spares requirements.

As noted earlier, we wish to evaluate the relative cost-effectiveness of METRIC, MOD-METRIC, VSL, AFLCR 57-6, in variants of these methods in managing Air Force Recoverable Items. Some of the interesting variations include the use of one of these techniques for initial provisioning and a second technique for decisions. For example, the Air Force currently uses MOD-METRIC for initial provisioning of major end items, while VSL is used for replenishment calculations. None of the above methods is currently used for distribution of Air Force recoverable spares, but work is currently underway to develop a distribution system based upon Sherbrooke's METRIC model.

To evaluate each of these alternate methods, we needed a set of computer programs to perform the millions of required calculations. To obtain the required computation capability, we modified the family of programs developed to implement the MOD-METRIC algorithm. By adjusting input data to these programs, and by making minor changes in the programs themselves, it was possible to compute stock levels using all 13 sets of inventory management rules that were of interest in this study.

Table IV-1 presents the relationship between the family of MOD-METRIC computer programs and the specific computational methods of interest in this study. As shown in the Table, with appropriate input data the MOD-METRIC program ONEIND can be used to perform stock level computations according to the assumptions of the original METRIC mathematical model. Similarly, the program TWOIND computes levels using Muckstadt's Mod-Metric mathematics. With

TABLE IV-1

CODES EMPLOYED TO SIMULATE
ALTERNATE INVENTORY MANAGEMENT METHODS

COMP CODE	COMPUTATIONAL METHOD	MOD-METRIC COMPUTER PROGRAM
1	METRIC	MOD-METRIC/ONEIND
2	MOD-METRIC	MOD-METRIC/TWOIND
3	VSL	MOD-METRIC/ONEIND, with all bases assumed equal, and with upper and lower bounds
4	AFLCR 57-27	R57-27, a modification of a CREATE Time Sharing program written by Mr. T. Mitchell, AFALD/XRS.

IMETH = Initial Provisioning Computation Code
KMETH = Replenishment Computation Code

slightly different manipulations of input data, Variable Safety Level (VSL) computations may be performed using the ONEIND program. Unfortunately, the family of MOD-METRIC computer programs provided no way of computing requirements using logic specified in AFLCR 57-27. However, Mr. Terry Mitchell of AFALD/XRS previously implemented a CREATE Time Sharing Program to accomplished this. We converted Mr. Mitchell's program to a subroutine to provide an AFLCR 57-27 computation capability in our Stock Levels Computation system.

To automate the levels computation process, it was convenient to assign codes to identify each of the basic stock level computation methodologies. These codes are shown on the left-hand side of Table IV-1. As shown in the table, the code IMETH is used to identify the computational methodology used for Initial Provisioning Calculations, while the code KMETH is used to specify the computational method for replenishment computations. Thus, IMETH = 1 indicates that the ONEIND program is to be used in initial provisioning calculations in accordance with METRIC math model assumptions, while KMETH = 1 indicates the same calculation is to be performed during the replenishment phase. Similarly, IMETH = 2 indicates that the TWOIND program is to be used to represent MOD-METRIC calculation methods in the initial provisioning phrase, while KMETH = 2 indicates TWOIND is to be used for replenishment calculations.

The computation codes IMETH and KMETH specify the general computational method to be used for Initial Provisioning and Replenishment Calculations, repsectively. However, three other pairs of codes are also used in RIME to completely specify a stock level computation method. These additional codes are defined in Tables IV-2 and IV-3.

TABLE IV-2
Levels Calculation Codes

NOTE: IMETH, IEQBAS, and ICOST apply to Initial Provisioning Calculations
KMETH, KEOQBAS, and KCOST apply to Replenishment Calculations

Fortran Variable	Description	Value	Effect	Calculation
IMETH, KMETH	Method Code	1	Use ONEIND Model for all items	Set IC1=IC2=1. (All items are treated as LRUs)
		2	Use TWOIND Model for all LRU/SRU groups	IC1=2 for LRU; IC1=3 for SRU. (LRU/SRU relationships are considered)
		3	Use ONEIND to determine qty, and EVALUATE to distribute assets	IC1=IC2= 1. (All items are treated as LRUs)
		4	Use AFLCR 57-27 logic and evaluate for all items	IC1=IC2 = 1 (All items are treated as LRUs)
IEQBAS, KEQBAS	Equal Bases Code	0	Treat Bases as read in	Base-to-base differences are recognized.
		1	Equal Base Assumption	Set output flying hours and order and ship times (FH(K) and OST(K)) equal to the average values for output
ICOST KCOST	Unit Cost Code	0	Unit Cost is Unchanged	Discount is not used.
		1	Set unit cost to (Unit Cost) (Discount)	Discount equals greater of .10 or (1.- NRTS fraction)

NOTE: IEQBAS and KEQBAS are used in program ONEIND only; no other program uses these variables.

TABLE IV-3
Stock Level Bounds Codes

Fortran Variable	Description	Value	Effect
IMINSK KMINSK	Bounds Code	0	No bounds are used.
		1	Set lower bound to expected number as assets in repair/resupply pipeline; set upper bound to BOMIN
BOMINI, BOMINK	Value of Upper Bound	.001	In ONEIND calculations, this variable specifies an upper bound on expected system backorders. Once system backorders are reduced to this value, no additional assets are allocated.
		.01	In TWOIND calculations, BSTOP is set to this value. BSTOP is the reduction in expected backorders per additional million dollars invested at which stock level allocations are stopped.

NOTE. IMINSK and BOMINI apply to Initial Provisioning Calculations
KMINSK and BOMINK apply to Replenishment Calculations

Table IV-2 defines two additional codes defining required manipulations of input data to the ONEIND and TWOIND programs. The codes IEQBAS and KEQBAS define whether or not base-to-base differences are to be recognized during stock levels computations. If the code equals 0, base-to-base differences are recognized. On the other hand, if the code equals 1 all bases are assumed to be equal. In the latter case, the stock level computations are performed by replacing input values for flying hour programs and order and ship times for each base by the respective average values for all bases combined. This latter data modification is required to represent the Equal Base Assumption employed in the VSL computation.

The codes ICOST and KCOST specify whether or not the cost discount computation utilized in VSL is to be employed. If this code equals zero, stock level computations are performed without any modification to the D041 unit cost. On the other hand, if the code equals 1, the D041 unit cost is multiplied by a discount factor which equals the greater of .10 or (1.-NRTS fraction). Code ICOST applies to initial provisioning predictions, while the code KCOST specifies the replenishment calculation method.

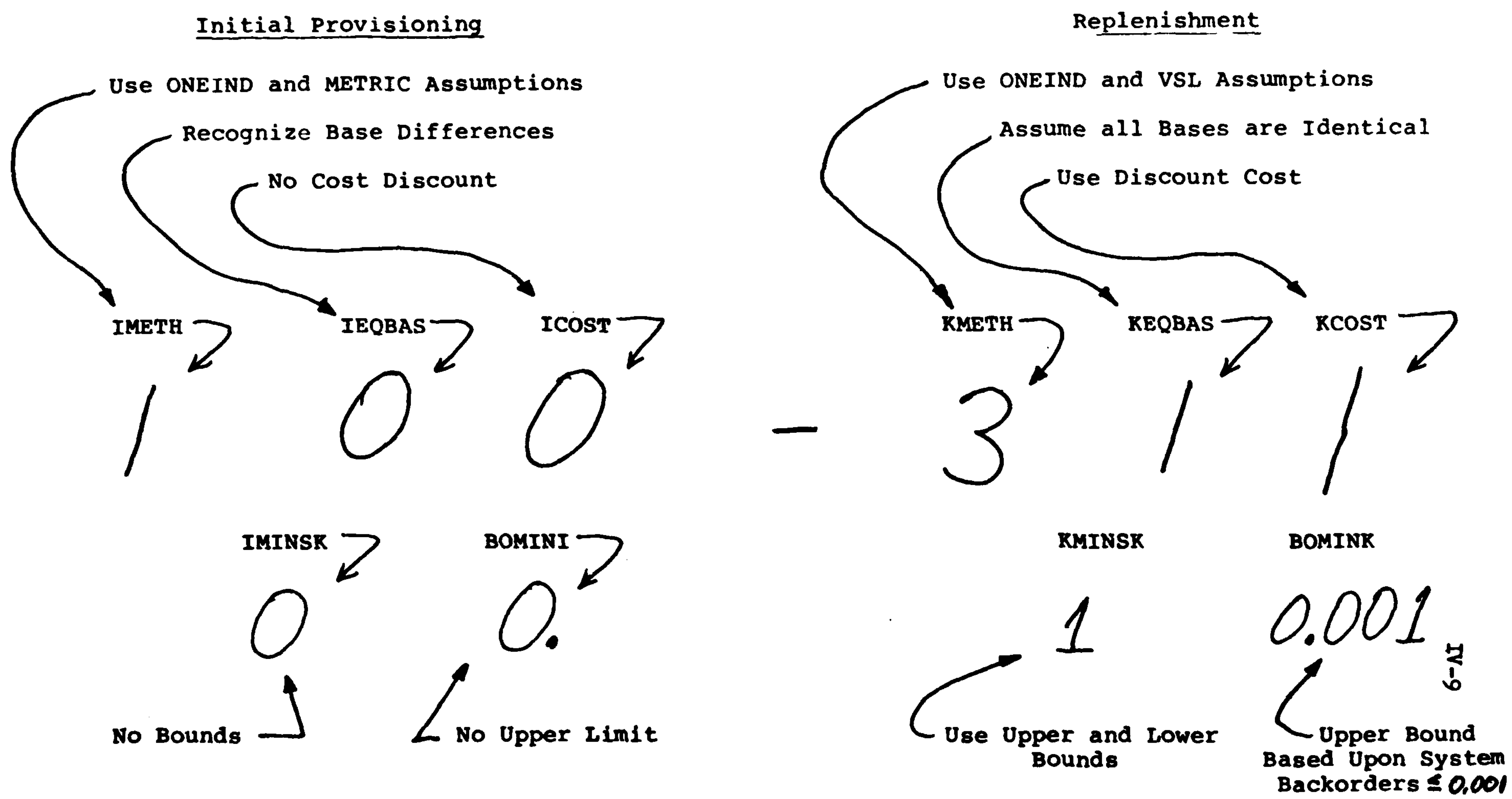
Table IV-3 specifies four variables used in establishing upper and lower bounds upon computed stock levels. The variables IMINSK and BOMINI are employed to specify bounds on initial provisioning calculations, while KMINSK and BOMINK specify values to be used in replenishment decisions. If the code IMINSK equals zero, no upper or lower bounds are used in the initial provisioning calculations. On the other hand, if the Bound code IMINSK is 1, all stock levels are computed with a lower bound equal to the expected number of assets in the repair/resupply pipeline, and an upper bound specified by Upper Bound Variable BOMINI. Specific numerical values for upper bounds used in this study and their interpretation are given in

Table IV-3. The above discussion applies to the variable IMINSK and BOMINI used to bound initial provisioning calculations. However, similar comments apply to the use of the variables KMINSK and BOMINK used to bound replenishment stock levels.

From the above discussion, ten numerical values are required to specify an inventory management policy for both initial provisioning and replenishment calculations. Five codes (IMETH, IEQBAS, ICOST, IMINSK, and BOMINI) are required to specify an inventory management policy for Initial Provisioning calculations, while five additional codes (KMETH, KEQBAS, KCOST, KMINSK, and BOMINK) are required to specify Replenishment calculations. For example, Figure IV-1 illustrates the inventory management code 100-311 0 0/1 0.001. As shown in the table, the first three digits (100) specify that the ONENID program is to be used to represent the METRIC computation algorithm; base-to-base differences are to be recognized; and finally, no cost discount factors are to be employed. The first set of bounds (0 0) specify that no upper or lower limits are to be used in stock level computations for initial provisioning. On the other hand, the replenishment code 311 indicates that the ONEIND program is to be used with Variable Safety Level assumptions. Since KEQBAS equals 1, all base variables are to be set to the average base values in these calculations. Further, since KCOST equals 1, a discounted unit cost is to be used. Finally, the set of bounds (1 0.001) indicate that both upper and lower limits are to be applied after the stock levels have been computed. The lower limit is to be the expected number of assets in the repair/resupply pipeline, while the upper bound on base stock levels is to be computed based upon minimum system backorders of 0.001.

FIGURE IV-1

Illustration of a Complete Inventory Management Code



SPNDMS

To simplify the submission of RIME evaluation runs, the program SPNDMS was written to automate the preparation of input data and Job Control Language statements. Thirteen specific combinations of inventory management codes are recorded in a table within the SPNDMS program. The codes included in this table are presented in Table IV-4. In utilizing SPNDMS, the program asks which of these 13 rules are to be used for inventory calculations. It then obtains the required parameters from the internal table, and generates the JCL required to implement the calculations. SPNDMS is discussed further in Chapter V; in the following sections, we discuss the job streams generated by SPNDMS.

Table IV-4

POLICIES EVALUATED

POLICY NO.	COMPUTE CODE	BOUND CODE	INITIAL PROVISIONING PROGRAM	EQUAL BASES	COST DISCOUNT	BOUNDS	REPLENISHMENT PROGRAM	EQUAL BASES	COST DISCOUNT	BOUNDS	DESCRIPTION
1	100-100	0 0/0 0	ONEIND	NO	NO	NO	ONEIND	NO	NO	NO	METRIC
2	310-310	0 0/0 0	ONEIND/ EVALUATE	YES	NO	NO	ONEIND/ EVALUATE	YES	NO	NO	METRIC WITH EQUAL BASES
3	101-101	0 0/0 0	ONEIND	NO	YES	NO	ONEIND	NO	YES	NO	METRIC WITH COST DISCOUNT
4	100-100	1.001/ 1.001	ONEIND	NO	NO	YES	ONEIND	NO	NO	YES	METRIC WITH BOUNDS
5	311-311	1.001/ 1.001	ONEIND/ EVALUATE	YES	YES	YES	ONEIND/ EVALUATE	YES	YES	YES	VSL
6	200-200	0 0/0 0	TWOIND	NO	NO	NO	TWOIND	NO	NO	NO	MOD-METRIC
7	200-200	1 .01/ 1 .01	TWOIND	NO	NO	YES	TWOIND	NO	NO	YES	MOD-METRIC WITH BOUNDS
8	400-311	0 0/ 1 .001	R57-27/ EVALUATE	N/A	N/A	N/A	ONEIND/ EVALUATE	YES	YES	YES	AFLCR 57-27 FOR INITIAL PROVISIONING. VSL FOR REPLENISHMENT. METRIC FOR DISTRIBUTION
9	200-311	0 0/ 1. 001	TWOIND	NO	NO	NO	ONEIND/ EVALUATE	YES	YES	YES	MOD-METRIC FOR INITIAL PROVISIONING. VSL FOR REPLENISHMENT BUYS. METRIC FOR DISTRIBUTION.
10	200-311	1 .01/ 1 .001	TWOIND	NO	NO	YES	ONEIND/ EVALUATE	YES	YES	YES	POLICY 9 WITH BOUNDS
11	400-100	0 0/ 1 .001	R57-27	N/A	N/A	N/A	ONEIND	NO	NO	YES	AFLCR 57-27 FOR INITIAL PROVISIOING. METRIC FOR DISTRIBUTION.
12	200-100	0 0/ 1 .001	TWOIND	NO	NO	NO	ONEIND	NO	NO	YES	MOD-METRIC FOR INITIAL PROVISION, METRIC FOR DISTRIBUTION
13	200-100	1 .01/ 1 .001	TWOIND	NO	NO	YES	ONEIND	NO	NO	YES	POLICY 12 WITH BOUNDS ON INITIAL PROVISIONING

Alternate Job Streams in the Levels Computation System

All Levels System job streams utilize a combination of MOD-METRIC programs and other programs specifically written for the RIME system. The MOD-METRIC programs ONEIND, TWOIND, GETBSO, and EVALUATE are essentially identifical to the AFLCR 57-6 MOD-METRIC programs with minor modifications required for the incorporation of upper and lower bounds. The programs SAVDAT, GETDAT, LEVLDP and SORT were specifically written for RIME. They provide for the storage and sorting of stock levels data produced by the MOD-METRIC routines.

Figure IV-2 illustrates the set of alternate job streams which might be utilized in runs of the Levels Computation System. The program DMSGN1 is a key element in this system for it generates the input records required by all the MOD-METRIC computer programs. The program DMSGN1 first obtains LRU and SRU descriptive data from the Master Data Set, and writes output records required as input to MOD-METRIC computational programs. Record layouts for data produced by DMSGN1 are presented in Section III of Appendix A.

As shown in Figure IV-2, records used to drive Initial Provisioning calculations are written to file I2, while Replenishment calculation input records are written to file R2. Branches on the left-hand side of Figure IV-2 correspond to Initial Provision Calculations, while those on the right-hand side of the figure correspond to Replenishment. The Calculation Codes IMETH and KMETH specify which of the specific branches will be followed in a given RIME run.

As shown in Table IV-2, setting Code IMETH = 1 indicates that the ONEIND computer program is to be used to simulate the computation of stock levels

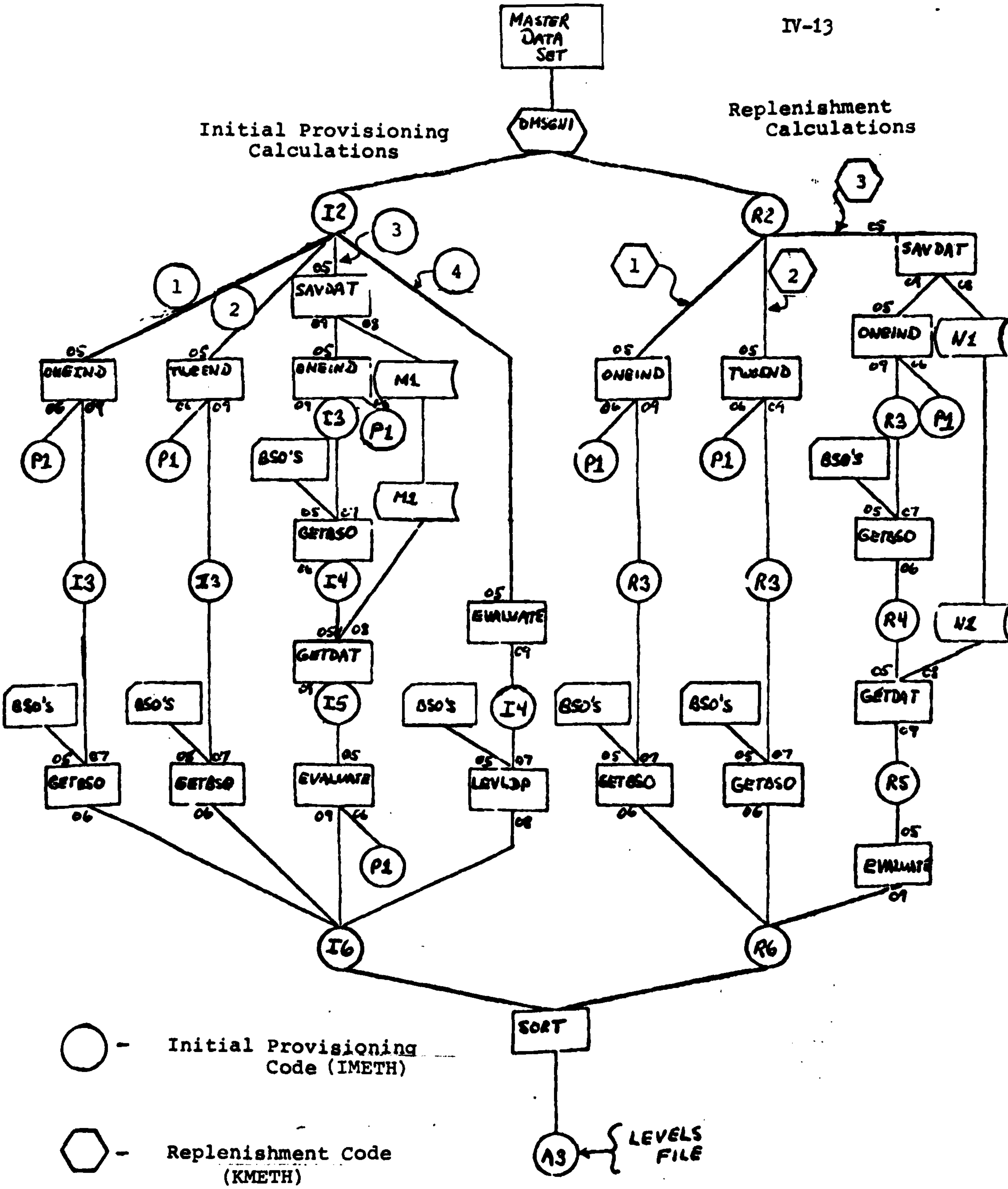


Figure IV-2. The Level Computation System.

according to the METRIC mathematical model. As shown in Figure IV-2, setting IMETH = 1 specifies that the job stream on the far left side of Figure IV-2 is to be used for Initial Provisioning Calculations. The first step in this process is the use of the program ONEIND to compute a set of stock levels for up to 20 different Buy Support Objectives (BSOs). Details of these calculations are printed to File P1, while the stock levels and associated buy suport objectives computed by ONEIND are output to File I3. File I3 serves as input to the MOD-METRIC program GETBSO. The GETBSO program reads specified buy support objectives from File 05, and scans the set of buy support objectives and their associated stock levels on input file I3. In performing this scan, the routine GETBSO attempts to find a set of stock levels with a buy support objective closest to the desired BSO. Once the required stock levels are found, GETBSO writes these levels to the output file I6.

Suppose that the Computation Code for replenishment calculations KMETH is also set to 1. As shown in Figure IV-2, this indicates that the MOD-METRIC program ONEIND is also to be used to compute stock levels for Replenishment calculations, while GETBSO is to be used to determine the specific stock levels associated with each desired buy support objectives. These levels are then output to File R6.

As shown in the figure, replenishment stock levels are always written to File R6. At the conclusion of a Levels Computation System run, Initial Provisioning stock levels on File I6 and Replenishment stock levels on File R6 are merged, sorted, and the resulting records are output to the "Levels File" A3. This file is one of major inputs to the RIME simulation model.

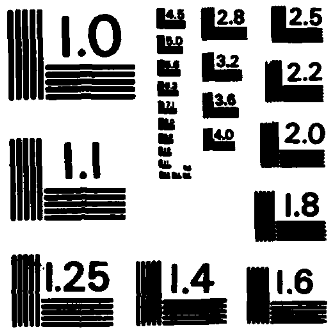
Before leaving this section, let us briefly discuss each of the other job streams illustrated in Figure IV-2. If the computation Code IMETH is set to 2, the

MOD-METRIC program TWOIND is used to compute stock levels according to Muskstadt's multi-indenture, multi-echelon mathematical model. This program is similar to the ONEIND program in that it computes stock levels for a range of buy support objectives. The set of levels computed by TWOIND is then written to the output file I3. In turn, file I3 serves as input to the program GETBSO. GETBSO then reads the desired buy support objectives from Logical Unit 05 and scans the levels file I3 to find a set of levels whose buy support objective is closest to the desired BSO. The resulting levels are output to File I6. Essentially the same calculations are performed when the replenishment Computation Code KMETH = 2.

AD-A121 981 RIME: THE RECOVERABLE ITEM MANAGEMENT EVALUATOR VOLUME 2/2
I MODEL DESCRIPTION(U) DECISION SYSTEMS DAYTON OH
W S DEMMY MAY 80 TR-80-01 F33600-78-C-0524

UNCLASSIFIED F/G 9/2 NL

END
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

As shown in Table IV-2, setting the Computation Code IMETH equal to 3 indicates that the Variable Safety Level (VSL) calculation is to be used in Initial Provisioning to determine the total number of assets to be procured, while the METRIC math model is to be used to determine the distribution of these assets. As shown in Figure IV-2, implementation of this computation requires several job steps. First, program SAVDAT is called. This program stores the MODMETRIC item identification and job control cards on the random file Ml. This file is used later in the job stream as input to the program GETDAT. After SAVSAT stores this informantion, the program ONEIND is used to compute stock levels. When IMETH =3, ONEIND performs these computations assuming that all bases are equal. Stock levels computed by ONEIND are output to file 13, which provides input to the program GETBSO. In turn, GETBSO determines the set of stock levels associated with the desired BSO's, and outputs these stock levels to file I4.

The program GETDAT reads the sets of stock level from file 14, and obtains unit cost, NRTS rate, failure rate, and other item decription data from the random file Ml. GETDAT then uses this information to construct a set of data cards for input to the MODMETRIC program EVALUATE. In doing this, GETDAT computes the total number of assets associated with each desired buy support objective by summing the stock levels for the depot and for each base. It then outputs a MODMETRIC "delivery schedule" card with the computed asset total in the first quarter deliveries field. The program EVALUATE then reads this total number of assets, and determines the optimum distribution of these assets among the depot and using bases. In performing its calculation, EVALUATE recognizes base-to-base LRU/SRU relationships.

Finally, stock levels computed by EVALUATE are output to File 16. If the Replenishment Computation Code KMETH = 3, calculations are similar to those for IMETH = 3, but in this case the computed stock levels are written to the file R6.

A final possibility is that the Initial Provisioning Computation Code IMETH is set to 4, indicating that logic specified in AFLCR 57-27 is to be used for Initial Provisioning calculations. When this option is specified, program DMSGN1 calls subroutine INTPRO to compute the total number of assets associated with AFLCR 57-27 logic. This total is recorded in the part number identification field of the MODMETRIC delivery schedule card is also output to file 12. Later in the job stream, program EVALUATE is used to determine distribution of the total number of AFLCR 57-27 assets among each of the using bases. Stock levels computed by EVALUATE are output to file 14.

The RIME Simiulation Model expects to find separate sets of stock levels for each Buy Support Objective to be evaluated. However, a desired buy support objective is not a factor in the AFLCR 57-27 computation logic, and AFLCR 57-27 logic always computes the same requirement regardless of the specified BSO. Consequently, the program LEVLDP is used to create duplicates of the AFLCR 57-27 stock levels output from EVALUATE. To do this. LEVLDP reads the BSO file, and counts the number of BSO records on this file. It then creates a like numver of duplicates of each stock level record on file 14, and writes all of these levels records to file 16. Stock levels on file 14 are then merged and sorted with the replenishment levels on file R6.

Note: Setting KMETH =4 is not a legal option in the Levels Computations System. That is, AFLCR 57-27 logic cannot be used in preforming replenishment calculations.

Chapter V

Using the Recoverable Item Management Evaluator

Introduction

To exercise the Recoverable Item Management Evaluator (RIME), the user must perform three major steps. These are:

1. Construct a Master Data Set containing required information on all LRUs and SRUs to be included in the evaluation calculations.

2. Utilize the Events Generator to create an exogenous event file to drive the Simulation Model.

3. Utilize the program SPNDMS.O to generate stock levels according to specified criteria, and to initiate a simulation run.

Let us now consider each of these steps in more detail.

Generating the Master Data Set

To utilize the Recoverable Item Management Evaluator the user must provide item description and historical reparable generation data for each LRU and SRU to be evaluated. Required formats for this information are documented in Appendix A of this report. Definitions of associated FORTRAN input variables used in the RIME system are defined as part of the record layouts presented in Appendix A. These data elements are designed to closely correspond to data elements in the D041 Recoverable Consumption Item Requirements System.

Using the Events Generator

The file EVTGN.A contains Job Control Language (JCL) required to exercise the Events Generator. A listing of EVTGN.A is shown in Figure V-1. The "SELECT" cards are used to select compiled object programs for each of the routines used by the Events Generator for use in the current run. Lines 210, 220, and 225 specify input parameters which control the EVTGN.A run.

Table V-1 defines the input parameters required as the first two lines of input to the EVTGN.A run. The first record specifies the set of LRU/SRU groups to be selected from the Master Data Set during this run, while the variable NBASES specifies the number of operating bases to be assumed during the events generation process. The variables INQTR and NREPL specify the number of quarters in the simulation planning horizon and the number of replications of the events generation process to be performed, respectively. For example, line 210 of Figure V-1 specifies that LRU/SRU groups 1 thru 3 are to be included in the current EVTGN.A run. In performing the events generation process, line 210 specifies that 6 bases are to be assumed, that the events generation process is to utilize a 16-quarter planning horizon and two replications of the events generation process are to be performed for each LRU/SRU group.

The second input line of EVTGN.A specifies a set of "Write" flags which specify output options to be employed during the current simulation run. These flags are defined in Table III-3. Finally, line 225 of Figure V-1 specifies an ASCII input file which contains base order and ship time information, as well as data describing flying hour programs by base for each period to be simulated. Table V-2 defines the variables which are specified by this file, while Table V-3 illustrates the structure of this file. As shown in Table V-2, the first line of this file specifies the number of bases, NBASES, for which data is provided in this file. In addition,

12/19/79   13.79

```
$      IDENT    WP1462,XRS/BEMMY          EVTGN.A
$      LIMITS   15,40K,,5K
$      OPTION   FORTRAN,NOMAP
$      NOTE     **********BEGIN RIME ROUTINES
$      SELECT   RIME/OBJ/EVTGN2.O
$      SELECT   RIME/OBJ/BASEDA.O
$      SELECT   RIME/OBJ/READF2.O
$      SELECT   RIME/OBJ/NOSET1.O
$      SELECT   RIME/OBJ/LRUEV2.O
$      SELECT   RIME/OBJ/SRUEVT.O
$      SELECT   RIME/OBJ/SRUIND.O
$      SELECT   RIME/OBJ/FELIST.O
$      NOTE     **********BEGIN REQS ROUTINES**********
$      SELECT   REQS/RANDU.O
$      EXECUTE
$      LIMITS   15,40K,,5K
$      DATA     05
  1  3   6   16   2    GRPS  1-3,  6-BASES, 16-QTRS, 2-REPLICATIONS.
  3 0 0 0 0   0 0 0 0 0   1 1 0 0 0    0 0 0 0 0
$      SELECTA  RIME/F111FH.D
$      TAPE     07,X1D,,74876,,D041EXT
$      TAPE     08,X2D,,72774,,RIME2/RING
$      ENDJOB
```

Figure V-1. EVTGN.A: Job Control Language for the Events Generator.

Table V-1

EVTGN.A Input Parameters

Record No.	Variables
1	NFGRP, NFGRP, NBASES, INQTR, NREPL
2	Write Flags (IWT(I), I = 1,20)

Definitions

NFGRP	=	The number of the first LRU/SRU group to be included in the Events Generation Process.
NLGRP	=	The number of the last LRU/SRU group to be included in the Events Generation Process.
NBASES	=	The number of operating bases assumed in events generation. This value overrides the number of bases specified in the Base OST/Flying Program File.
INQTR	=	The number of quarters of events to be generated (INQTR must not exceed 16).
NREPL	=	The number of replications of the event generation process to be performed for each LRU/SRU group.
IWT(I)	=	Write Flag I. See Table III-3 for definitions of output options controlled by IWT(I).

Table V-2

Base O&ST and Flying Program Input File

Record No.	Data Elements
1	LINE, NBASES, (OSTDLT(K), K = 1, NBASES)
2	LINE, IDATE, (BFH(I,K), K = 1 NBASES)
3,4,5. . .	Information in the format of Record #2 must be provided for each quarter to be simulated, in ascending time sequence.

Definitions

LINE	=	A Time-Sharing File line number.
NBASES	=	The number of operating bases associated with this data file.
OSTDLT(K)	=	The deviation of the average depot order and ship time at base K from the worldwide average depot order and ship time.
IDATE	=	The date associated with flying program data on this input line. This variable is not used in any calculations.
BFH(I, K)	=	Flying program at base K in quarter I.

Table V-3.

A Sample Base Data Input File.

100	6	-6	-6	9	9	-2	-4
110	741	2908	2908	2140	2140	5261	5755
120	742	2618	2618	2063	2063	4763	5841
130	751	2816	2816	2273	2273	4769	5234
140	752	2618	2618	2197	2197	5412	6151
150	753	2179	2179	2259	2259	5501	6440
160	754	1135	1135	2048	2048	4626	4907
170	761	2465	2465	2097	2097	4853	6276
180	762	690	690	1311	1311	4906	4622
190	763	566	566	1598	1598	4488	6185
200	764	1736	1736	1489	1489	4158	5282
210	771	2320	2320	2356	2356	4826	5462
220	772	2804	2804	2325	2325	5526	4313
230	773	1785	1785	2700	2700	5431	4546
240	774	1828	1828	2207	2207	5863	5060
250	781	1619	1619	1351	1351	2748	3280
260	782	1738	1738	1543	1543	3621	3146
270	783	2514	2514	2470	2470	5292	4534
280	784	2230	2230	2329	2329	4773	4281
290	791	1820	1820	1963	1963	4789	4549
300	792	2367	2367	2216	2216	5622	5874
310	793	2400	2400	2500	2500	5700	4900
320	794	2400	2400	2500	2500	5500	5000
330	801	2350	2350	2600	2600	4700	4800

the first line of this file also specifies values for computing the Order and Ship Time (O&ST) for an individual base once the worldwide average OST value is known. The remaining lines in the file specify the flying program activity by base for each of the quarters to be simulated.

In Figure V-1, line 230 specifies the data tape which contains the Master Data Set, while line 240 specifies the data tape that is to contain the Exogenous Event File. This latter tape then becomes the major input element to the RIME Simulation Model.

Using SPNDMS.O to Generate RIME Simulation Runs

To exercise the Levels Computation System and to initiate an associated simulation run, it is necessary to:

1. Provide appropriate input parameters to the program DMSGN1 which specify the specific Initial Provisioning and Replenishment calculations to be performed, and

2. Create an appropriate set of Job Control Language commands to bring the required levels of computation programs into use, and to control input/output files which interconnect these programs. This is a substantial task, since several hundred JCL cards may be required for certain types of stock levels computations. Fortunately, many of the required input parameters remain constant in performing a given study of proposed inventory management methods. This was true in our case. Consequently, we developed the program SPNDMS to automate a major portion of the run specification task.

The Levels Computation System input data generator DMSGN1 requires two major types of input data. One of these is the Master Data Set discussed above.

The other major input is a set of parameters which specify the specific stock leveling calculations to be performed in a given run. Table V-4 defines these required input parameters. The FORTRAN variable names used in program DMSGN1 are shown in Table V-4. All of these values are read using a Free-Field input format.

As shown in Table V-4, the first two input lines to DMSGN1 specify the Computation Codes needed to define required Initial Provisioning and Replenishment calculations. The third input record specifies the set of LRU/SRU groups to be included in the current run, the number of quarters for which levels are to be computed, and a value of the variable NDHIS. The fourth input line specifies the set of Write Flags to be employed in the current run. Definitions of these DMSGN1 Write Flags are presented in Table V-5.

Input lines 5 thru 7 specify values to be used in MOD-METRIC computer program control cards. These parameters are defined in AFLCR 57-6, and will not be discussed further here. Finally, the remaining input cards to DMSGN1 specify the base Order and Ship Time calculation variables and flying hour programs by base to be used for the current computation run. These latter inputs are the same as those defined in Table V-2.

Fortunately, it is not necessary for the user to specify these data cards each time he wishes to exercise the RIME system. Rather, the program SPNDMS may be used to automate most of this process.

Using SPNDMS

The compiled program SPNDMS.O is a FORTRAN Time-Sharing program which automates many of the steps required in the preparation of input data and associated JCL statement required in the use of the RIME model. Figure V-2

Table V-4

DMSGN1 Input Parameters

Input Record							
1	IDENT	IMETH	IEQBAS	ICOST	KMETH	KEQBAS	KCOST
2	IMINSK	BOMINI	KMINSK	BOMINK			
3	NFGRP	NLGRP	INQTR	NDHIS*			
	WRITE FLAGS		(IWT(I),I = 1,20)				

MOD-METRIC CONTROL CARD DATA (See AFLCR 57-6 for definitions of these MOD-METRIC variables)

5	NQTR	KPO	CFAC9			
6	NOUT	IPRNT	IPNCH	IBSO		
7	NBIS	BETA	BSTART	BSTOP	CFAC	PBINC

BASE DATA

(See Table V-2 for description of required parameters.

NOTE: All DMSGN1 input parameters are read by FREE-FIELD format statements.

*NDHIS = The number of data periods to be considered as past history in performing levels calculations at simulation time zero. performing levels calculations. For most RIME applications, NDHIS should be zero.

Table V-5

DMSGN1 Write Flag

Write Flag Number	Routine	Effect
1	READFL	Input data print flag, where 1 = print short header record only 2 = print short header and rep gen data 3 = print all input data
2	READFL	Print item data written to Exogenous Event File (In DMSGN 1, this flag should always equal zero.)
3	DMSGN1	Print intermediate data generation calculations
4	METINP	Print details of MTBD, NRTS, and condemnation rate estimates.
5	INTPRO	Print initial provisioning calculations
6	OVHSTL	Print overhaul stock level calculations.
7-20	-	These flags are not used.

illustrates the use of this program for initiating a particular RIME simulation run. As may be seen in the Figure, SPNDMS.O asks the reader several questions. Answers to Question #1 define where outputs from the current run are to be directed and the size of the runs to be performed. Questions #2 and #3 define the JCL control card which identifies the Exogenous Event File to be used as input to the Simulation Model. Question #4 asks for the values of NFGRP, NLGRP, INQTR, and NDHIS to be provided as input to DMSGN1. It also asks for the maximum amount of CPU time to be used in the current run specified in hundredths of CPU-hours. Finally, Question #5 asks for the formula number to be evaluated in the current RIME run. Values for the 13 Computation Codes required to specify Initial Provisioning and Replenishment calculations defined in Table IV-4 are recorded in a table within the SPNDMS program. After the user specifies the required formula number, SPNDMSdoes a table lookup to determine the values for the variables IMETH, IEQBAS, and so on, and SPNDMS then prints these values at the user's terminal. Next, SPNDMS accesses the specific files of JCL statements required to implement the desired calculations. Once the set of required input parameters in Job Control Language statements have been accumulated, SPNDMS submits the entire set of JCL records to the CREATE batch processing system. The CREATE control number assigned to the batch job is then printed at the user's terminal. In Figure V-2, the job number g365t was assigned.

In Question #6, SPNDMS asks if any more evaluation runs are to be launched for the current set of LRU/SRU groups. If the user answers yes, program logic returns to Question #5. Otherwise, SPNDMS asks Question #7. Question #7 asks if the user wishes to continue generating RIME evaluation runs, or if he wishes to terminate the current session with SPNDMS. If the user wishes to continue, program logic returns to Question #1. Otherwise, SPNDMS terminates its run.

```
 RUN SPNDMS.O
AC? PUNCH? SIMULATE? MORE CORE? (1=YES,  0=NO        ←Question #1*
=1  0       1         0
 IS --$:TAPE:07,X9D,,70053,,EXOGF                   ←Question #2
 OK FOR EXOGFILE? (Y=YES)
=N
  INPUT EXOGFILE CONTROL CARD                       ←Question #3
=$:TAPE:07,X9D,,72774,,RIME2
  CARD WAS READ AS--
=====$:TAPE:07,X9D,,72774,,RIME2
 IS CARD OK?(Y OR N)
=Y                                                  ←Question #4*
    NFGRP,    NLGRP,INQTR,NDHIS, CPU-LIMIT
=     1          3     16    0      15


                                                    ←Question #5
IDENT?
=3
    IDENT  IMETH  IEQBAS  ICOST  KMETH  KEQBAS  KCOST
     3      1       0      1      1       0       1
   IMINSK BOMINI  KMINSK  BOMINK
     0     0.        0     0.
  SNUMB # s365t                                     ←Question #6
CONTINUE?(Y OR N?)
=N                                                  ←Question #7
 DO ANOTHER GROUP?(Y OR N)
=N
```

***NOTE:** See Table V-6 for further description of these questions.

Figure V-2. An Illustration of the Use of SPNDMS.O to Initiate a RIME Evaluation Run.

Table V-6

SPNDMS.O Input Parameters

Question #1 (1 = Yes, 0 = No)

AC?	Is the output for this job to be directed to station code AC? If not, output is printed at the CREATE Central Site.
PUNCH?	Are the computed stock levels to be output on punched cards?
SIMULATE?	Is the RIME Simulation Model to be used (1 = YES). Otherwise, the run terminates after all stock levels have been computed.
MORE CORE?	Does the current run involve more than 40 Stock Keeping Units for the largest LRU/SRU family to be considered? If so, more core is needed.

Question #4

NFGRP	=	The number of first LRU/SRU group to be included in calculation.
NLGRP	=	The number of the last LRU/SRU group.
INQTR	=	The number of quarters in Planning Horizon
NDHIS	=	The number of quarters of input data to be considered as past history at simulation time zero. For most RIME applications, set NDHIS = 0.
CPU-LIMIT	=	Job time limit for the current run in hundredths of CPU-hours.

RIME Simulation Model Input/Output Files

Figure V-3 illustrates the input and output files utilized by the RIME Simulation Model, and the associated Logical Unit Designators. As shown in the figure, Logical Units 05, 07, and 09 provide the inputs to the Simulation Model, while logical units 06, 15, 16, and 43 are output files. Logical unit 11 is a work file used for the temporary storage of stock level data.

Logical Unit 05 specifies run parameters and output options to be used in the control of the current simulation run. Logical Unit 07 is associated with the Exogenous Event File, while Logical Unit 09 is associated with the set of Stock Levels to be evaluated during the current simulation run. The Stock Levels File is a sequencial file. To simplfy processing, this file is read in sequential order by subroutine INITM1 at the beginning of a RIME run, and stock levels read from this file are recorded on the random file 11. Stock levels are then obtained as needed by subroutine LEVEL from file 11 during the simulation run. This greatly simplifies the processing of mutiple replication evaluation runs.

The Simulation Model provides outputs to report codes 06, 15, and 16, and also provides punched card results to logical unit 43. The Summary and Short-Form Reports are printed to file 06, while files 15 and 16 are used to print 8- and 16-quarter totals of selected statistics. The information printed on files 15 and 16 is also punched on logical unit 43 if the flag ITWRT = 2.

Details of the Exogenous Event File and the Stock Levels File have been discussed in previous sections, and will not be considered further here. In the following sections, we discuss the features of the Run Parameter file, and of the output options available to users of the RIME Simulation Model.

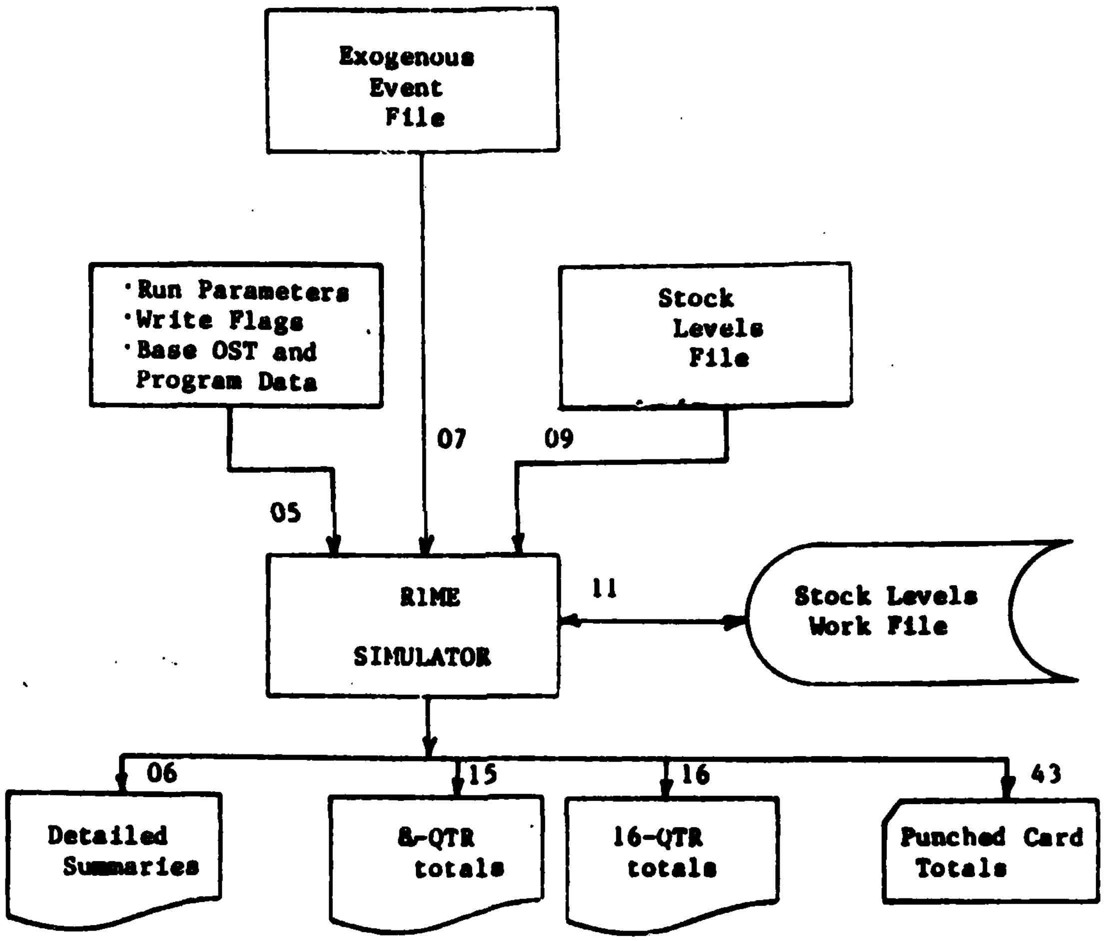


Figure V-3. Rime Simulation Model Input and Output Files.

Run Parameters File

The Run Parameters File (Logical Unit 05) specifies variables which control the size of the current simulation run, and the output products to be produced as a result of this run. Table V-7 defines the variables specified in this file, while Figure V-4 illustrates a "print-back" of Run Parameter data produced in a sample Simulation Model run.

As shown in Table V-7, the first set of Run Parameters are specified in the header record of the Exogenous Event File (Logical Unit 07). This record specifies the set of LRU groups to be included in the current run, and the number of bases, the length of the planning horizon and the number of replications for which exogenous events are provided. This record is created automatically during the Events Generator run. The remaining Run Parameters are provided in free-field format on Logical Unit 05. The first record on Logical Unit 05 specifies four Output Controls, while the second record specifies values for 7 "debug" flags. The Debug flags are useful in the development of new programs for the Simulation Model. Finally, the third input record specifies the size of the simulation run to be performed. Definitions of these parameters are presented in Table V-4.

RIME Output Products

As discussed in Chapter II, all RIME performance statistics have three different indices. For example, the Performance Statistic IREQT(I, J, K) records the total number of requisitions received from customers of the supply organization. The index I denotes the quarter in which the requisition was received. The index J records the type of measure associated with this statistic, where J = 1 denotes the number of distinct federal stock numbers or distinct actions associated

Table V-7

Simulation Model Control Parameters

(C1) Exogenous Event File Characteristics

Note: The following variables are read from the header record of the Exogenous Event File (EEF) on Logical Unit 07.

NFGRP	=	The number of the first LRU group on the EEF.
NLGRP	=	The number of the last LRU group on the EEF.
NBASES	=	The number of bases assumed in the EEF.
NNQTR	=	The number of quarters of events on the EEF.
NREPL	=	The number of replications performed for each LRU group.

(C2) Output Controls. . . (Note. 1 = YES)

ITWRT	=	The item "write" flag. If ITWRT = 1 or 2, subroutine ITRSLT called to punch ten major statistics for each replication of each LRU group. If ITWRT = 1, cummulative values through quarters 8 and INQTR, respectively, are punched. If ITWRT = 2, the statistics are punched for each of the INQTR quarters simulated.

Table V-7 (Continued)

IOUT = Detailed Summary Flag. If this flag equals 1, the Detailed Summary Report is produced for each Buy Support Objective (BSO) evaluated. This option produces approximately 1800 lines of output for each BSO.

IGRAPH = The Plot Flag. This option is not implemented in the current version of RIME.

ISUMRY = The Short Form Summary Flag. If this parameter equals 1, the Short Form Summary Report is produced after for Buy Support Objective evaluated.

(C3) Debug Flags.

Note: These variables specify print options useful in debugging new RIME routines.

IDBUG = The File Debugging Flag. If IDBUG = 1, the following actions are taken:

(a) All entries and removals from the Future Events List are printed.

(b) All entries and removals from the Work-in-Process file are printed.

(c) All entries and removals to the Backorder File are printed.

Table V-7 (Continued)

(d) Values of on-hand, on-order, work-in-process stock, and backorders are printed after each event.

(e) The Stock Status Trace Report is produced.

IEBUG	=	The Item Input Data Flag. If IEDUG = 1, all LRU/SRU item description data input from file 07 is printed.
IFBUG	=	The Item Forecast Flag. If IFBUG = 1, details of endogenous item forecasting calculations are printed. (In the current version of RIME, forecasting calculations are exogenous events. Consequently, this flag has no effect.)
IGBUG	=	Statistic Update Flag. If IGBUG = 1, the results of statistics updates in routines CUM and CUMB are printed.
IHBUG	=	Levels Calculations Flag. If IHBUG = 1, the results of all stock levels calculations are printed.

(C8) Simulation Size Parameters.

NLAM	=	The number of Buy Support Objectives (LAMBDAS) to be evaluated.
INQTR	=	The number of quarters to be simulated. This number must not exceed the value NNQTR read from the header record of the Exogenous Event File.
NTOTL	=	The total number of LRU groups to be simulated in the current run. This value must not exceed (NLGRP - NFGRP + 1).

INVENTORY SYSTEM SIMULATOR

RUN PARAMETERS

(C1) EXOGENOUS EVENT FILE CHARACTERISTICS

NFGRP	FIRST LRU GRP =	1
NLGRP	LAST LRU GRP =	1
NBASES	NUMBER OF BASES =	6
NQUTR	NUMBER OF QUARTERS =	16
NREPL	NUMBER OF REPLICATIONS	1

(C2) OUTPUT CONTROLS...(NO=0, 1=YES)

ITWRT	IT.WRITE= 0 **
IOUT	SUMMARY = 1
IGRAPH	GRAPHS = 1 **
ISUMRY	SUMMARY = 1

(C3) DEBUG FLAGS

IDBUG = 1

IEBUG = 1

IFBUG = 1 **

IGBUG = 0

IHBUG = 0

ITRACE= 0

ISTRAC= 20

(C4) SIMULATION SIZE

NLAM	NUMBER OF LAMBDAS	1
NQRTR	NUMBER OF QUARTERS	16
NTOTL	NO. OF LRU GROUPS	1

**Note: This option is not implemented in the current version of RIME.

Figure V-4. RIME Simulation Model Control Parameters.

with the current event, J = 2 denotes the number of units associated with the event, and J = 3 denotes the dollar value of all units associated with the event. Finally, the index K denotes the aggregation category for the statistic. Values of K and their meaning are as follows:

Aggregation Index Value	Definition
1	LRU at base level
2	SRU at base level
3	LRU at the depot
4	SRU at the depot
5	SRU at the Aircraft Overhaul Facility
6	SRU at the Aircraft Overhaul Facility

For example, suppose that a requisition for 12 units of a $10 item is received at a base during the fourth quarter of the simulation. In this case, the period index is I = 4. Consequently, the RIME statistic IREQT(4, 1, 1) is increased by 1 to record this order action. Since 12 units were ordered, the statistic IREQT(4, 2, 1) is increased by 12, and the statistic IREQT(4, 3, 1) is increased by 120 to record the dollar value of the requisition.

As discussed in Chapter II, RIME collects 32 different statistics, for 16 distinct quarters. To print all of the statistics collected in a single replication of the RIME model results in a very large volume of output.

<u>Detailed Summary Report</u>. If the input parameter IOUT = 1, RIME prints the Detailed Summary Report to Logical Unit 06. This report prints the values of every RIME performance statistic, an output of approximately 1800 lines for each Buy Support Objective evaluated. Figures V-5 thru V-8 illustrate the format of this report. As shown in Figure V-5 thru V-7, three pages of output are produced for each value of the aggregation index K. Figure V-5 illustrates the first of these three pages for an LRU at a base facility (aggregation index K = 1). This page displays the receipts, returns, shipments, ordering actions, and requisitions associated with LRUs at base level during the simulation run. As shown in the figure, these statistics are given by quarter, and results are shown both in terms of number of actions/FSNs involved, the number of units associated with these actions, and the dollar values of these units.

Figure V-6 illustrates the second page of the three page output associated with the aggregation index K = 1. This page records the number of expediting, rationing, disposal, and termination actions that were recorded in the simulation run, as well as the number of reparable generations, condemnation, and NRTS actions perfromed. The final columns of this page record the number of repairs completed within each quarter, the number of assets that in a work-in-process status at the end of each quarter, and the total number of days that assets spent waiting for parts.

Figure V-7 illustrates the third page of statistics output for the aggregation index K = 1. This page records the supply performance observed during the simulation run. Values for end of period backorders, backorder days, inventory weeks, and observed fill rates by priority class are displayed on this page of the report.

LRU AT BASE FACILITY

NGROUP = 1 NREPL = 1 NLAK = 1

PERIOD	(1) INVENTORY ON-HAND	(2) INVENTORY ON-ORDER	(3) RECEIPTS	(4) RETURNS	(5) TOTAL SHIPMENTS	(6) PRI 1 SHIPMENTS	(7) ORDERS PLACED	(8) TOTAL REQS	(9) REQS CANCELED	(10) PRIORITY 1 REQS
*** ACTIONS/FSN A ***										
1	2	0	7	0	113	113	5	114	0	114
2	2	0	0	0	126	126	6	132	0	132
3	2	0	0	0	108	108	0	102	0	102
4	1	0	2	0	98	98	2	103	0	103
5	1	1	5	0	127	127	6	126	0	126
6	1	0	3	0	113	113	2	111	0	111
7	1	0	4	0	136	136	4	137	0	137
8	0	2	6	0	146	146	8	140	0	140
9	1	0	2	0	137	137	0	134	0	134
10	2	0	1	0	177	177	1	178	0	178
11	1	1	1	0	142	142	2	148	0	148
12	2	1	2	0	116	116	2	111	0	111
13	2	0	1	0	164	164	0	164	0	164
14	3	0	2	0	138	138	2	136	0	135
15	2	0	0	0	67	67	0	85	0	86
16	1	0	1	0	112	112	1	112	0	112
TOTALS	24	5	37	0	1980	1980	35	1980	0	1980
AVE/YR	7.	1.	9.	0.	495.	495.	9.	495.	0.	495.
*** UNITS ***										
1	2	0	7	0	113	113	5	114	0	114
2	2	0	0	0	126	126	0	132	0	132
3	2	0	0	0	108	108	0	102	0	102
4	1	0	2	0	98	98	2	103	0	103
5	1	1	5	0	127	127	6	126	0	126
6	1	0	3	0	113	113	2	111	0	111
7	1	0	4	0	136	136	4	137	0	137
8	0	2	6	0	146	146	8	140	0	140
9	1	0	2	0	137	137	0	134	0	134
10	2	0	1	0	177	177	1	178	0	178
11	1	1	1	0	142	142	2	145	0	145
12	2	1	2	0	116	116	2	111	0	111
13	4	0	1	0	164	164	0	164	0	164
14	3	0	2	0	138	138	2	136	0	136
15	2	0	0	0	67	67	0	86	0	86
16	1	0	1	0	112	112	1	112	0	112
TOTALS	26	5	37	0	1980	1980	35	1980	0	1980
AVE/YR	7.	1.	9.	0.	495.	495.	9.	495.	0.	495.
$$$ DOLLARS $$$										
1	17300	0	43050	0	694950	694950	30750	731100	0	781100
2	12300	0	0	0	774...	[illegible]	[illegible]	811800	0	811800
3	12300	0	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0	827300
4	6150	0	12300	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0	633450
5	6150	6150	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	0	774900

Figure V-5. Sample LRU Summary Report, Page 1.

Figure V-5

Column Definitions

Column	Statistics	Definition
(1)	INVOH	Inventory on hand at the end of the quarter
(2)	INVOR	Total quantity on order at the end of the quarter
(3)	IRECET	Receipts of replenishment orders
(4)	IRETRN	Serviceable returns from customers of the supply system
(5)	ISHIPT	Total shipments to fill new customer requisitions or to fill backorders
(6)	ISHIPI	Total shipments to fill priority 1 requisitions or backorders
(7)	IORDER	Total replenishment orders initiated
(8)	IREQT	Total requisitions received
(9)	IREQC	Total outstanding backorders that are cancelled by the customer
(10)	IREQI	Total priority 1 requisitions received

LRU AT BASE FACILITY

NGROUP = 1 PRIPL = 1 NLAM = 1

PERIOD	(11) EXPEDITES	(12) RATIONING ACTIONS	(13) DISPOSALS	(14) TERMIN	(15) REP FENS	(16) CONDEMNS	(17) NRTS	(18) COMPLETED REPAIRS	(19) WORK IN PROCESS	(20) WAIT FOR PARTS
*** ACTIONS/FSN A ***										
1	0	0	0	0	115	0	5	100	1	9
2	0	0	0	0	131	0	0	120	3	14
3	0	0	0	0	102	0	0	100	1	3
4	0	0	0	0	103	0	2	95	4	11
5	0	0	0	0	126	0	6	122	4	7
6	0	0	0	0	111	0	2	110	3	7
7	0	0	0	0	138	0	4	132	3	14
8	0	0	0	0	145	0	8	139	4	5
9	0	0	0	0	135	0	0	136	3	9
10	0	0	0	0	177	0	1	177	2	5
11	0	0	0	0	148	0	2	140	4	16
12	0	0	0	0	111	0	2	115	1	6
13	0	0	0	0	107	0	0	105	2	5
14	0	0	0	0	134	0	2	135	1	4
15	0	0	0	0	85	0	0	86	0	1
16	0	0	0	0	112	0	1	110	1	5
TOTAL	0	0	0	0	1900	0	35	1944	37	123
AVE/YR	0.	0.	0.	0.	475.	0.	9.	486.	9.	31.
*** UNITS ***										
1	0	0	0	0	115	0	5	100	2	4845
2	0	0	0	0	131	0	0	126	7	6540
3	0	0	0	0	102	0	0	100	1	1665
4	0	0	0	0	103	0	2	95	7	5973
5	0	0	0	0	126	0	6	122	5	3705
6	0	0	0	0	111	0	2	110	4	3505
7	0	0	0	0	138	0	4	132	6	7011
8	0	0	0	0	145	0	8	139	4	2165
9	0	0	0	0	135	0	0	136	3	4135
10	0	0	0	0	177	0	1	177	2	2475
11	0	0	0	0	148	0	2	140	6	6969
12	0	0	0	0	111	0	2	115	2	3170
13	0	0	0	0	107	0	0	105	4	2275
14	0	0	0	0	134	0	2	135	1	1830
15	0	0	0	0	85	0	0	86	0	335
16	0	0	0	0	112	0	1	110	1	2365
TOTAL	0	0	0	0	1900	0	35	1944	57	58703
AVE/YR	0.	0.	0.	0.	475.	0.	9.	486.	14.	14676.
$$$ DOLLARS $$$										
1	0	0	0	0	158238	0	33758	664200	12308	28566758
2	0	0	0	0	665670	[illegible]	[illegible]	[illegible]	43059	40221888
3	0	0	0	0	627336	[illegible]	[illegible]	[illegible]	[illegible]	9870753
4	0	0	0	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
5	0	0	0	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Figure V-6. Sample LRU Summary Report, Page 2.

Figure V-6

Column Definitions

Column	Statistic	Definition
(11)	IEXPED	Total expediting actions initiated
(12)	IRATON	Total of all rationing actions within the period
(13)	IDISPS	Total disposals within the period.
(14)	ITERM	Total terminations; i.e., totals for all replenishment orders that are cancelled or reduced when the inventory position exceeds the termination level
(15)	IREPGN	Total number of reparable generations in the current period
(16)	ICNDEM	Total number of reparable generations condemned in the current period
(17)	INRTS	Total number of NRTS actions taken in the current period If the stocking location is a base or the Aircraft Overhaul Facility, this column represents assets which are transported from that location to the depot. If the stocking location represents the depot, this column represents NRTS assets received within the current period.
(18)	IRECPL	Total number of assets which completed repair within the current period.
(19)	IWIP	Total number of assets which are in a work-in-process status at the end of the period.
(20)	IWFP	Total number of days spent waiting for parts in the current period. This time is not recorded until an LRU is removed from the wait-for-parts status.

LRU AT BASE FACILITY

NGROUP = 1 NREPL = 1 NLAM = 1

PERIOD	(21) TOTAL BACKORDERS	(22) PRIORITY I BACKORDERS	(23) TOTAL NO-DAYS	(24) PRI I NO-DAYS	(25) INVENTORY WEEKS	(26) TOTAL FILLS	(27) PRIORITY I FILLS	(28) TOT FILLS /TOT REQ	(29) PRI I FILLS /PRI I REQS
			*** ACTIONS/FSN A ***						
1	1	1	50	50	19	48	48	0.35	0.35
2	7	7	71	71	14	35	35	0.27	0.27
3	1	1	35	35	24	76	76	0.69	0.69
4	6	6	61	61	17	46	46	0.45	0.45
5	5	5	55	55	20	48	48	0.38	0.38
6	3	3	41	41	14	40	40	0.36	0.36
7	4	4	73	73	8	46	46	0.34	0.34
8	4	4	35	35	11	76	78	0.53	0.53
9	1	1	42	42	14	47	47	0.35	0.35
10	2	2	55	55	24	67	67	0.38	0.38
11	8	8	84	84	14	64	64	0.43	0.43
12	3	3	65	65	15	48	48	0.43	0.43
13	3	3	32	32	26	72	72	0.69	0.69
14	1	1	30	30	17	99	99	0.73	0.73
15	6	6	15	15	25	64	64	0.74	0.74
16	3	8	46	46	19	54	54	0.48	0.48
TOTALS	49	49	784	784	281	918	918		
AVE/YR	12.	12.	196.	196.	70.	230.	230.	0.464	0.464
			*** UNITS ***						
1	1	1	50	50	19	48	48	0.35	0.35
2	7	7	71	71	14	35	35	0.27	0.27
3	1	1	35	35	24	76	76	0.69	0.69
4	6	6	61	61	17	46	46	0.45	0.45
5	5	5	55	55	20	48	48	0.38	0.38
6	3	3	41	41	14	48	48	0.36	0.36
7	4	4	73	73	8	46	46	0.34	0.34
8	4	4	35	35	12	78	78	0.53	0.53
9	1	1	42	42	15	47	47	0.35	0.35
10	2	2	55	55	24	67	67	0.38	0.38
11	8	8	84	84	14	64	64	0.43	0.43
12	3	3	65	65	15	48	48	0.43	0.43
13	3	3	32	32	28	72	72	0.69	0.69
14	1	1	30	30	17	99	99	0.73	0.73
15	6	6	15	15	26	64	64	0.74	0.74
16	8	8	48	48	19	54	54	0.48	0.48
TOTALS	49	49	784	784	286	918	918		
AVE/YR	12.	12.	196.	196.	72.	230.	230.	0.464	0.464
			$$$ DOLLARS $$$						
1	8150	8150	387900	387900	116850				
2	43650	43650	436650	436650	86100				
3	8150	8150	219250	219250	147[illegible]				

Figure V-7. Sample LRU Summary Report, Page 3.

Figure V-7

Column Definitions

Column	Statistic	Definition
(21)	IBACKT	Total backorders outstanding at the end of the period.
(22)	IBACKI	Total priority 1 backorders outstanding at the end of the period.
(23)	IBAKDT	Total backorder-weeks observed during the period. For example, if one item has a single requisition for 12 units in a backorder status for 3 weeks, there are three requisition-weeks of backorders, and three X 12 = 36 unit-weeks of backorders for that item.
(24)	IBAKDI	Total backorder-days observed for priority 1 requisitions.
(25)	INVDAY	Total number of inventory-weeks observed. If there are 15 units on hand for the first 8 weeks of a quarter, and 6 units on hand for the remaining four weeks, a total of (15 X 8 + 6 X 4) = 144 inventory unit-weeks were observed in the period.
(26)	IFILLT	Total number of requisitions that were filled "off-the-shelf, i.e., that were filled without backordering.
(27)	IFILLI	Total fills for priority 1 requisitions.
(28)	IFILLT/ IREQT	Fill rate for the period based upon the totals of all requisitions received.
(29)	IFILLI/ IREQI	Priority 1 fill rate.

As noted above, the aggregation index K may take on six distinct values, and three separate statistics pages are produced for each of these indices -- a total of 18 pages of output statistics for each Buy Support Objective. Figure V-9 illustrates the first of the three output statistics pages associated with an aggregation index K = 2. This index records results associated with SRUs at base level. This report has the same format as for the case when K = 1, and will not be discussed further here.

Short-Form Summary Report. Because of the very large volume of printout associated with the Detailed Summary Report, the user will usually find it desirable to develop a specialized output report designed for the needs of the specific project in which he is involved. Figure V-9 illustrates the format of a report which might be developed in this effort. This figure displays totals of six major categories of performance statistics. The report displays the number of requisitions which a given stocking location submits to suppliers of that location, the total amount of time spent waiting for parts, the number of backorder weeks observed in filling requisitions submitted to that supply organization, and the total number of requisitions which customers of the organization submitted. The total number of these requisitions which were filled off the shelf, and the corresponding fill percentage is also displayed. Separate statistics are printed for each value of the aggregation index K, as well as totals for each category. Figure V-8 illustrates statistics expressed in units (J = 2); however, the Short-Form Report has three pages, one page for each of the three possible values of the measure type index J.

Quarter Totals and Punched Card Output. To facilitate statistical analysis of RIME simulation results, the values of 10 major statistics are punched on cards by subroutine ITRSLT at the conclusion of each BSO evaluation replication. Record

SRU AT BASE FACILITY

NGROUP = 1 NRLPL = 1 NLAP = 1

PERIOD	INVENTORY ON-HAND	INVENTORY ON-ORDER	RECEIPTS	RETURNS	TOTAL SHIPMENTS	PRI I SHIPMENTS	ORDERS PLACED	TOTAL REQS	REQS CANCELED	PRIORITY I REQS
*** ACTIONS/ESN A ***										
1	2	1	9	0	9	0	8	10	0	0
2	2	1	12	0	14	0	12	14	0	0
3	2	0	3	0	3	0	2	2	0	0
4	2	0	12	0	12	0	12	11	0	0
5	1	2	5	0	7	0	7	8	0	0
6	2	0	7	0	7	0	5	7	0	0
7	2	0	13	0	14	0	13	13	0	0
8	2	1	5	0	5	0	6	6	0	0
9	2	0	7	0	9	0	6	8	0	0
10	2	0	5	0	5	0	5	5	0	0
11	2	0	15	0	16	0	15	16	0	0
12	2	0	8	0	8	0	8	8	0	0
13	2	1	5	0	5	0	6	6	0	0
14	2	0	4	0	4	0	3	3	0	0
15	2	0	1	0	1	0	1	1	0	0
16	2	0	5	0	5	0	5	5	0	0
TOTALS	31	6	116	0	124	0	114	123	0	0
AVE/YR	8.	2.	29.	0.	31.	0.	29.	31.	0.	0.
*** UNITS ***										
1	2	1	9	0	9	0	8	10	0	0
2	2	1	12	0	14	0	12	14	0	0
3	2	0	3	0	3	0	2	2	0	0
4	2	0	12	0	12	0	12	12	0	0
5	1	2	5	0	7	0	7	8	0	0
6	2	0	7	0	7	0	5	7	0	0
7	2	0	13	0	14	0	13	13	0	0
8	2	1	5	0	5	0	6	6	0	0
9	2	0	7	0	9	0	6	8	0	0
10	2	0	5	0	5	0	5	5	0	0
11	2	0	15	0	16	0	15	16	0	0
12	2	0	8	0	8	0	8	8	0	0
13	2	1	5	0	5	0	6	6	0	0
14	2	0	4	0	4	0	3	3	0	0
15	2	0	1	0	1	0	1	1	0	0
16	2	0	5	0	5	0	5	5	0	0
TOTALS	31	6	116	0	124	0	114	124	0	
AVE/YR	8.	2.	29.	0.	31.	0.	29.	31.		
$$$ DOLLARS $$$										
1	984	452	4068	0	4068	0	3[illegible]			
2	984	452	5424	0	6328	0				
	984	0	1356	0	1356					
	984	0	5424	0	5424					
	452	984	2260							

Figure V-8. Sample SRU Summary Report, Page 1.

SIMULATION RESULTS IN UNITS (J=?)

	(1)	(2)	(3)	(4)	(5)	(6)
	REQUISITIONS TO SUPPLIERS	WAIT FOR PARTS	BACKORDER WEEKS	TOTAL REQUISITIONS	TOTAL FILLS	FILL PERCENTAGE
BASE LRU	35	55763	784	1960	918	0.468
BASE SRU	114	0	77	124	15	0.121
TOTAL	149	55763	861	2084	933	0.448
DEPOT LRU	15	0	0	35	35	1.000
DEPOT SRU	13	0	0	116	116	1.000
TOTAL	28	0	0	151	151	1.000
OVRHL LRU	0	0	0	0	0	0.
OVRHL SRU	2	0	0	2	2	1.000
TOTAL	2	0	0	2	2	1.000
GRAND TOT	179	55763	861	2237	1086	0.485

Figure V-9. Sample Short-Form Report.

Code F1 in Appendix A defines the format of these output cards, and the FORTRAN variable names used to produce them. In addition, the punch card data is also printed to Logical Units 15 and 16. As shown in Figure V-10, the report printed to Logical Unit 15 displays totals of activities for the first 8 quarters of the simulation for depot buy-dollars, depot backorders, base fills, base requisitions, and base backorder-days for both LRUs and SRUs. The first column of Figure V-10 presents the number of the Buy Support Objective used in the stock level computations, while the second column of the figure displays the replication number associated with the output statistics. Observe that the LRU and SRU base requisitions (IREQT) are identical in each of the 10 runs shown in Figure V-10. This is because the Exogenous Events Generator forces an identical number of reparable generations in each replication of the simulation model. Note, however, that the buy dollars and other support statistics change for varying values of MLAM and different replications, reflecting increasing buy dollars and support effectiveness for increasing values of MLAM, and varying support performance within replications for a given value of MLAM.

Figure V-11 displays the report printed to Logical Unit 16. This report is identical in format to that of Figure V-10. However, this report provides the totals of statistics accumulated over the entire 16 quarter simulation period.

Debugging Aids. As noted above, the user may specify values for several debug flags to assist in the development of new RIME routines. Figure V-12 illustrates outputs obtained when the flags IEBUG and IDBUG are set to 1.

IEBUG. When IEBUG = 1, all item data input from the Exogenous Event File is printed. A sample of the printout is shown at the top of Figure V-12. The specific variables printed correspond exactly to the variable names shown for the Item Data

8-QTR TOTALS
NGROUP = 9 ICENT = 9

		DEPOT ICRDLR-S		DEPOT ISODAT		BASE IFILLT		BASE IBEGT		BASE ISODAT	
	I	LRU	SRU	LRU	SRU	LRU	SRU	LRU	SRU	LRU	SRU
1	1	1483200	2073035	2346	107826	9	1047	246	1623	18923	18777
1	2	1453200	2041486	2381	106279	6	1083	246	1623	17905	16931
2	1	1483200	2073035	2346	107826	9	1047	246	1823	18923	18777
2	2	1483200	2041486	2381	106279	6	1083	246	1623	17905	16931
3	1	2060000	2842501	4067	127759	20	1394	246	1623	13467	8568
3	2	2060000	2612486	4000	126031	25	1419	246	1823	12665	7025
4	1	3213600	4124350	5955	200696	53	1824	248	1823	9574	6818
4	2	3213600	4094074	5888	197992	67	1862	246	1623	7620	4060
5	1	3213600	4153356	6287	199850	54	1817	246	1623	8833	6488
5	2	3213600	4119753	6277	197228	61	1863	246	1823	7460	4447

Replication Number

MLAM, the Buy Support Objective Number

Figure V-10. Logical Unit 15 Report.

16-OIP TOTALS
NGROUP = 9 IDENT = 9

		DEPOT IORDER-S		DEPOT IBODAY		BASE IFILLT		BASE IRPRT		DIST IBCDAY	
	I	LRU	SRU	LRU	SRU	LRU	SRU	LRU	SRU	LRU	SRU
1	1	1483200	2090086	2346	114845	52	2616	608	3512	28877	21576
1	2	1483200	2067544	2359	113523	53	2667	607	3507	29120	20552
2	1	1483200	2090086	2346	114845	52	2616	608	3512	28877	21576
2	2	1483200	2067544	2361	113523	53	2667	607	3507	29120	20552
3	1	2069000	2896593	4067	133839	113	3200	608	3512	19444	9008
3	2	2060000	2874937	4006	132726	133	3216	607	3507	18725	7865
4	1	3213600	4145193	5955	217776	234	3284	608	3512	12547	6939
4	2	3213600	4122873	5888	215143	241	3307	607	3507	11654	4862
5	1	3296000	4174199	6287	217635	237	3289	608	3512	11827	6834
5	2	3296000	4148747	6277	215232	249	3310	607	3507	10548	4958

Replication Number

MLAM, the Buy Support Objective Number

Figure V-11. Logical Unit 16 Report.

Record (Record Code B2) presented in Appendix A. Basically, these variables display unit cost and time delay values associated with each LRU and SRU.

IDBUG. When IDBUG = 1, an information line is printed whenever events are entered or removed from the Future Events List, or whenever there are changes in the backorder or Work-in-Process Files. As illustrated in Figure V-12, when IDBUG = 1 the subroutine INGASP prints the dimensions used to initialize the Work-in-Process file. The next six lines show details of event transactions being placed on the F.E.L. by subroutine INITAL through calls to subroutine ENTER. The list of events on the F. E. L. at time 0 is also printed when IDBUG = 1.

The above discussion concerned the entry of event notices onto the F.E.L. The bottom of Figure V-12 illustrates the information printed when an event is removed from the F.E.L. As shown in the figure, at time 0, a type 13 event is removed from the F.E.L., while at time 1 (ITIME = 1), an Initial Provisioning Event (ITYPE = 20) is removed from the F.E.L.

Stock Status Trace Report. When IDBUG = 1, the Stock Status Trace Report is produced on logical unit 52. The first 5 columns of this report display the event time (ITIME), the event type (ITYPE), and the three event parameters (IP1, IP2, and IP3). The remaining columns display the status of stock on hand, on order, work-in-process, and backordered at the time each event is removed from the F. E. L. These values are displayed by Stock Keeping Unit. Values for the first 8 SKUs are displayed on the first line, and any additional SKU values are displayed in subsequent groups of 8 on following lines.

ITRACE, ISTRAC. At times the analyst will not want to produce the many lines of print that result when IDBUG = 1, although he may be interested in

IEBUG=1

```
*****BEGIN CRU GRP=   1
  2    1    0  ISEQ=       1  FSN=5841062475357    UCOST=    6157.00
                   JUPA  JNDTT  JBRT  JBRT  JDORT  JTADM  JIPROD  JBDURD  JUVSHP  JSHIP1,2,....,8
                     2     27     0     3     8     90    270      1       0        2  2 17 17  6  4  0  0
  3    1    0  ISEQ=       2  FSN=5841604899855    UCOST=    452.20
                   JUPA  JNDTT  JBRT  JRPT  JDORT  JTADM  JIPROD  JBDURD  JUVSHP  JSHIP1,2,....,8
                     2     27     5     6     5     90    180      1       8        2  2 17 17  6  4  0  0
  0    0    0  ISEQ=       3  FSN=EVENTS FOLLOW     UCOST=    0.
                   JUPA  JBDTT  JBRT  JRRT  JDORT  JTADM  JIPROD  JBDURD  JUVSHP  JSHIP1,2,....,8
                     0     0      0     0     0     0      0       0       0        0  0  0  0  0  0  0  0
*****INITEM.  NITEM=  16     NSRC=    1
```

IDBUG=1

INITIAL

```
*****INITAL.....INITIALIZE FEL,GASP,NJ-FILES,AND EVENTS
*****INGASP--ITIME=        0
                   MAX FILE ENTRIES             100
                   NUMBER OF FILES                2
                   DIMENSION OF NSET            600
                   NO. OF ATTRIBUTES              4
          ITIME    MAX FILE ENTRIES             600     JTIME   JTYPE   JP1   JP2   JP3    FILE INDEX
ENTER       0                                           6430      6      0     0    6400      1      1    1   8430
ENTER       0                                              1     20      0     0      0       2      2    2     1
ENTER       0                                            690     11      1     0      0       3      2    3     1
ENTER       0                                         134460     18      0     0      0       4      2    4     1
ENTER       0                                              0     13      0     0      0       5      5    5     6
ENTER       0                                             20     13      0     0      0       6      5    6     6
```

Parameters of Work-in-Process File

INITEM2

```
BEGIN INITM2....ITEM INITIALIZATION
*****EXIT INITM2
```

```
                   FUTURE EVENTS LIST AT TIME        0
                              NFINST=        5   NENTRY=       6
        N     JTIME    JTYPE    JFSN    JQTY    JPRIOR   JPOINT   FILE INDEX
        1        0      11        0       0       0        2          5
        2        1      20        0       0       0        6          2
        3       20      13        0       0       0        3          6
        4      690      11        1       0       0        1          3
        5     6430       6        0       0    7400        4          1
        6   134406      18        0       0       0        0          4
```

F.E.L. At Time=0

```
                  NO MORE ENTRIES ON THE FUTURE EVENTS LIST
#1 → REMOVE        0      13      0      0      0      5
     ITIME   ITYPE     J     QTY    IPS     N    INVACT INVDUF INNIP NBUID NBOTU NBOIR NBUTR NTIME JTIME
       0      13       0      0      6                                                      1     21
#2 → REMOVE        1      20      0      0      0      2
     ITIME   ITYPE     N     QTY    IPS     N    INV...                NBUIU NBOIR NBUTR NTIME JTIME
       1      20       0      0      0                                                     20     21
     *****REVIEW--NS   1  UN=     0  OJL-IN...
     ENTER             1
     *****...=   1  IBL=   12
```

Figure V-12. Sample Debug Outputs.

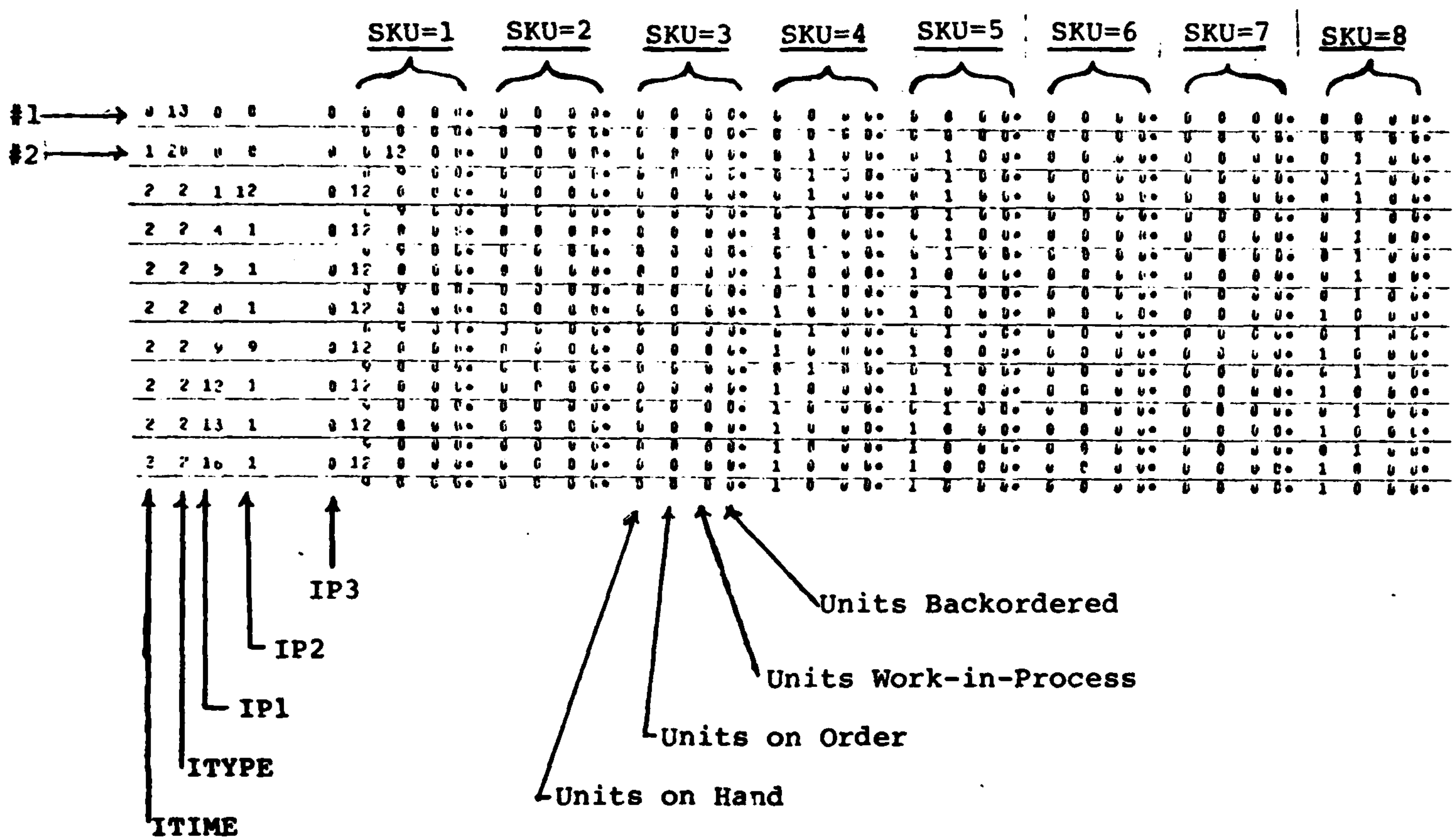


Figure V-13. Sample Stock Status Trace Reprot.

obtaining event printouts during a specific time interval in the simulation. The flags ITRACE and ISTRACE provide this capability. The variable ITRACE specifies the point in simulated time that a detailed trace of simulation of events is the start. At that point in time, the flag IDBUG is set equal to 1. This causes the event by event printout to start. The flag remains equal to 1 until the simulated clock time ISTRAC. At this time, IDBUG is reset to zero, and the event by event printouts cease.

REFERENCES

1. Demmy, W. Steven, RIME: The Recoverable Item Management Evaluator, Volume I: Model Description, TR-80-01, Decision Systems, 3575 Charlene Drive, Dayton, Ohio 45432, May 1980, 153 pp.

2. Demmy, W. Steven, RIME: The Recoverable Item Management Evaluation Volume II, Program Listings and Narratives, TR-80-02, Decision Systems, 3575 Charlene Drive, Dayton, Ohio 45432, May 1980, 298 pp.

3. Demmy, W. Steven, An Empirical Evaluation of Proposed Stockage Policies for Recoverable Item Management, TR-80-03, Decision Systems, 3575 Charlene Drive, Dayton, Ohio 45432, May 1980, 173 pp.

4. Demmy, W. Steven and Victor J. Presutti, Jr., Multi-Echelon Inventory Theory in the Air Force Logistics Command, Working Paper 76-3011-27, Department of Management, Wright State University, Dayton, Ohio 45435.

5. Muckstadt, John A., "A Model for a Multi-Item, Multi-Echelon, Multi-Indenture Inventory System, Management Science, v20, n4, December, 1973, pp. 472-481.

Appendix

RIME Record Layouts

RIME

Record Types

		Record Code
A.	Master Data File	
	Item Description Record	A1
	Past Usage Record	A2
B.	Enogenous Event File	
	File Identification Record	B1
	Item Data Record	B2
	Exogenous Event Record	B3
C.	MOD-METRIC Input Files	
	Work Unit Code, Part Number Format	C0
	Item Data Record	C1
	Delivery Schedule Record	C2
	Stock/Number Data Record (not used in RIME)	C3
	Title/Comment Data Format (Not used in RIME)	C4
	GETBSO Parameter Record	C5
	EVALUATE Parameter Record	C6
	Flying Hour and OST Record	C7
	Output Control Record	C8
	Computation Parameter Record	C9
D.	GETBSO BSO/Stock Levels File	
	BSO Summary Record	D1
	BSO Levels Record	D2
E.	Levels Files	
	ITEM LEVELS RECORD	E1
F.	RIME Results File	
	LRU Group Results Record	F1

INPUT /OUTPUT RECORD	A1	SYSTEM NUMBER RIME	PAGE 1 OF 3 PAGES A1

TITLE RIME ITEM DESCRIPTION RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS
1.* ~ Dimensioned by item number N,
2. RECORD IS READ BY READFL

NO.	DATA ELEMENT NAME	FORTRAN FORMAT	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	SEQUENCE NUMBER	I6	ISEQ	1-6	6	N
2	DATA TYPE IDENTIFIER-1 (2 ⇒ LRU, 3 ⇒ SRU)	I1	IC1	7	1	N
3	DATA TYPE IDENTIFIER-2	I1	IC2	8	1	N
4	RECORD TYPE	I2	IREC	9-10	2	N
5	ALC SITE CODE	A2	ALC	11-12	2	A
6	SUB-GROUP MASTER STOCK No.	A15	FSN	13-27	15	AN
7	ITEM NAME (10)	A10	NAME	28-37	10	AN
8	UNIT PRICE	F9.2	UCOST*	38-46	9	N
9	EXPENDABLE, RECOVERABLE, REPARABLE CATEGORY	A1	ERRC	47	1	A
10	UNIT OF ISSUE	A2	UI	48-49	2	A
11	ADMINISTRATIVE LEADTIME MONTHS	I1	LTADM*	50	1	N
12	PRODUCTION LEADTIME MONTHS	I2	LTPROD*	51-52	2	N
13	BASE REPAIR CYCLE DAYS (3POS)	I3	IBRT*	53-55	3	N
14	DEPOT REPAIR CYCLE DAYS	I3	~~IDRCD~~	56-58	3	N
15	BUDGET PROGRAM CODE	A2	BPCODE	59-60	2	AN
16	SYSTEM MANAGEMENT CODE	A4	SMCODE	61-64	4	AN
17	MATERIAL PROGRAM CODE (MPC)	A4	MPCODE	65-68	4	AN
18	O & ST DAYS	I2	IOST*	69-70	2	N
19	O I M STOCK LEVEL DAYS	I2	IOIMSL	71-72	2	N
20	NJR STOCK LEVEL DAYS	I2	NJRSL	73-74	2	N
21	DEPOT FLOATING STOCK LEVEL	I3	IDFSL	75-77	3	N

INPUT /OUTPUT RECORD	SYSTEM NUMBER	A1 PAGE 2 OF 3 PAGES

TITLE	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
RIME ITEM DESCRIPTION RECORD				
DATA RECORD LENGTH / TAPE RECORD LENGTH/ BLOCKING FACTOR				
REMARKS				

NO.	DATA ELEMENT NAME	FORTRAN FORMAT	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
22	TOTAL OIM DMD RATE CURR USED	F5.4	OIMDR*	78-82	5	N
23	SRC ~~Total OIM DEMAND RATE CURR YR~~	~~A1~~			1	A
24	BASE PROCESSED PCT CURR USED	F3.2	RTSF*	83-85	3	SN
25	PDM NON JOB ROUTED REPAIR %	F3.2	PDMF*	86-88	3	N
26	EOH NON JOB ROUTED REPAIR %	F3.2	EOHF*	89-91	3	N
27	MISTR NON JOB ROUTED REPAIR %	F3.2	[illegible]*	92-94	3	N
28	OIM DEPOT DEMAND RATE	F5.4	DPDDR*	95-99	5	N
29	BASE NRTS PERCENT CURR USED	F3.2	BNRTSF	100-102	3	N
30	PDM JR CONDEMNATION PCT	F3.2	PDRCND*	103-105	3	N
31	PDM NJR REPLACEMENT %	F3.2	PNERC*	106-108	3	N
32	OIM BASE REPAIR RATE CURR USED	F5.4	OIMBR*	109-113	5	N
33	BASE CONDM PERCENT CURR USED	F3.2	BCONF*	112-116	3	N
34	EOH JR CONDEMNATION %	F3.2	EJCNF*	117-119	3	N
35	EOH NJR REPLACEMENT %	F3.2	ENCNF*	120-122	3	N
36	DEPOT OH CONDM PCT CURR USED	F3.2	DOHCNF*	123-125	3	N
37	MISTR JR CONDEMNATION %	F3.2	RMSCNF*	126-128	3	N
38	MISTR NJR REPLACEMENT %	F3.2	RMNRF*	129-131	3	N
39	COUNT OF APPLICATIONS	I5	NAPPL	132-136	5	N
40	NUMBER OF USERS	I3	NUSRS	137-139	3	N
41	USAGE END DATE	I4	IENDT*	140-143	4	N
42	ITEM ESSENTIALITY CODE	A3	IESSN	144-146	3	AN
43	UNIT REPAIR COST	F9.2	RCOST*	147-155	9	N
44	NOT USED	IX				

AFLC FORM MAR 72 484 REPLACES AFLC FORM 609, WHICH WILL BE USED. AFLC—WPAFB—APR 72 92M

3

INPUT /OUTPUT RECORD	SYSTEM NUMBER	A1 PAGE 3 OF 3 PAGES

TITLE	INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE
RIME ITEM DESCRIPTION RECORD				
DATA RECORD LENGTH / TAPE RECORD LENGTH/ BLOCKING FACTOR				
REMARKS				

NO.	DATA ELEMENT NAME	FORTRAN FORMAT	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
45	UNIT REPAIR MANHOURS	I6	IRPHRS	156-161	6	N
46	JR STOCK LEVEL DAYS	I2	JRSLD	162-163	2	N
47	BASE PROCESSING DAYS	I3	IBPD*	164-166	3	N
48	REPARABLE INTRANSIT DAYS	I3	IRINTR*	167-169	3	N
49	SOR TO MAINTENANCE DAYS	I3	ISORM	170-172	3	N
50	SHOP FLOW DAYS	I3	ISFLOW	173-175	3	N
51	SERVICEABLE TURN IN DAYS	I3	ISRTIN	176-178	3	N
52	REPAIR LEADTIME DAYS	I3	IRLT	179-181	3	N
53	COMP CYCLE FOR THIS DATA					
54	STARTING DATE	I4	IPBD	182-185	4	N
55	ITEM QPA	I4	IQPA	186-189	4	N
56	APPLICATION PERCENTAGE	I3	IAP	190-192	3	N

INPUT /OUTPUT RECORD	A2	SYSTEM NUMBER RIME	PAGE 1 OF 1 PAGES A2

TITLE: RIME PAST USAGE RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS: RECORD TYPES 03→11

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
1	SEQUENCE NUMBER	ISEQ	1-6		I6
2	DATA TYPE IDENTIFIER (2 ⇒ LRU, 3 ⇒ SRU)	IC1	7		I1
3	DATA TYPE IDENTIFIER-2	IC2	8		I1
4	RECORD TYPE*	IREC	9-10		I2
5	FILLER	-	11-12		2X
6	PAST USAGE IN QTR I, I=1,2,...,16		13-108		16I6
	03- BASE REP GENS	IBRGN**			
	04- BASE RTS	IRTS**			
	05- BASE CONDEMNATIONS	IBCOND**			
	06- TOTAL BASE NRTS	INRTS**			
	07- TOTAL DEPOT REP GENS	IDRGN**			
	08- TOTAL DEPOT CONDEMNATIONS	IDCOND**			
	09- TOTAL DEPOT REPAIRED	IDREP**	— wild 3 Sep 79		
	10- TOTAL OVERHAUL CONDEMNED	IOVCND**			
	11- PAST INSTALLED PROGRAM (K hours) (In 100's)	IPPROG**			
	** EACH OF THESE VARIABLES ARE DIMENSION (N,K), N= ITEM NUMBER K= QUARTER NUMBER				

INPUT /OUTPUT RECORD	B1	SYSTEM NUMBER RIME	B1 PAGE 1 OF 3 PAGES

TITLE: Exogenous Event File Identification Record

DATA RECORD LENGTH:

TAPE RECORD LENGTH/ BLOCKING FACTOR:

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

REMARKS

1. Data is binary
2. Record is built by EVTGN2; and read by RIME

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	Number of First LRU Group	NFGRP			
	Number of Last LRU Group	NLGRP			
	Number of Bases	NBASES			
	Number of Quarters	NNQTR			
	Number of Replications	NREPL			

6

INPUT /OUTPUT RECORD	B2	SYSTEM NUMBER RIME	B2 PAGE 2 OF 3 PAGES

TITLE: EXOGENOUS EVENT ITEM DATA RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS
1. DATA IS BINARY
2. RECORD IS BUILT BY READFL2; AND READ BY INITEM.

NO.	DATA ELEMENT NAME	RIME VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	DATA TYPE IDENTIFIER 1	IC1			I
	(NOTE; IC1 = 0 DENOTES END-OF-ITEM DATA				
	IC1 = 2 DENOTES AN LRU				
	IC1 = 3 DENOTES AN SRU)				
	DATA TYPE IDENTIFIER 2	IC2			I
	RECORD TYPE (MUST = 1)	IC3			I
	SEQUENCE NUMBER	ISEQ			I
	FEDERAL STOCK NUMBER				A15
	UNIT COST				F9.2
	QUANTITY PER APPLICATION	IQPA			I
	BASE-TO-DEPOT TRANSIT TIME (DAYS)	IBDTT*			I
	DEPOT REPAIR TIME (DAYS)	IDRT			I
	BASE REPAIR TIME (DAYS)	IBRT*			I
	OVER-HAUL JOB-ROUTED REPAIR TIME (DAYS)	IDORT ⟹	NOT USED IN PRESENT VERSION		
	PROCUREMENT ADMINISTRATIVE LEAD-TIME (DAYS)	LTADM			I
	PROCUREMENT PRODUCTION LEAD TIME (DAYS)	LTPROD			I
	BASE-TO-DEPOT ORDER TIME (DAYS)	IBDORD*			I
	DEPOT-TO-OVERHAUL FAC. SHIP TIME (DAYS)	IOVSHP			I
	DEPOT-TO-BASE SHIP TIME (DAYS), BASE 1,2,...	IDSHIP			I

*RIME USES MOD-METRIC ASSUMPTION THAT THESE TIMES ARE IDENTICAL FOR ALL BASES.

INPUT /OUTPUT RECORD	83	SYSTEM NUMBER RIME	83 PAGE 3 OF 3 PAGES

TITLE: EXOGENOUS EVENT RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS

1. DATA IS BINARY

NO.	DATA ELEMENT NAME	RIME VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	EVENT TIME	JTIME			I
	EVENT TYPE	JTYPE			I
	PARAMETER #1 (USUALLY= SKU No)	JP3			I
	" #2 (USUALLY = QTY)	JP4			I
	" #3 (MEANING VARIES)	JP5			I

NOTES

1. A TYPE 10 EVENT (END-OF-RUN) MUST TERMINATE EVERY EVENT SET FOR EVERY REPLICATION. ONLY ONE TYPE 10 EVENT PER REPLICATION IS PERMITTED.

INPUT /OUTPUT RECORD	CO	SYSTEM NUMBER	PAGE CO OF PAGES

TITLE: RIME METRIC RECORD IDENTIFICATION FIELDS

DATA RECORD LENGTH | TAPE RECORD LENGTH/ BLOCKING FACTOR

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

REMARKS: THESE DATA ELEMENTS REPLACE THE WORK UNIT CODE & PART NO. FIELDS OF MOD-METRIC INPUT RECORDS.

NO.	DATA ELEMENT NAME	RIME FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	INVENTORY MGT. POLICY NUMBER	IDEN	4-5		I2
	λ- NUMBER		6-7		I2
	LRU GROUP NUMBER		8-9		I2
	QUARTER NUMBER		10-11		I2
	ITEM NUMBER (LRU = 1, SRU#1 = 2, ETC)		12-13		I2
	RECORD SEQUENCE NUMBER	ISEQ	14-16		I3
	BLANK FIELD		17-24		8X
	OVERHAUL FACILITY STOCK LEVEL		25-27		I3

NOTE:

In Rine, the above field definitions are used in place of the Work Unit Code/Part Number Fields defined on the MOD-METRIC System Record Layouts. This convention applies to all of the MOD-METRIC input records.

C1

RECOVERABLE ITEM DATA INPUT FORMAT

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (See Below)	IC1	I1	1	N
"	IC2	I1	2	N
Blank		1X	3	
Work Unit Code	IDLRU IDSRU(J)	A24	4-11	A/N
Part Number			12-27	A/N
Use Code (See Figure 3-2-10)	UC SUC(J)	A1	28	A
Cost Source Code (See Figure 3-2-10)	PSD SPSD(J)	A1	29	A
MTBD Source Code (See Figure 3-2-10)	SD SSD (J)	A1	30	A
Unit Cost	CLRU CSRU(J)	F 7.0	31-37	N
Mean Time Between Demands (Hours)	YMTBD XMTBD(J)	F 6.0	38-43	N
Not Reparable This Station (Fraction)	YNRTS XNRTS(J)	F 4.2	44-47	N
Condemnation Rate (Fraction of NRTS)	CONL CONS(J)	F 3.2	48-50	N
Quantity per Next Higher Assembly	APP SAPP(J)	I2	51-52	N
Base Repair Time (Days)	BRTLRU BRTSRU(J)	F 3.0	53-55	N
Depot Repair Time (Days)	DRTLRU DRTSRU(J)	F 3.0	56-58	N
Procurement Lead Time (Months)	PLT SPLT(J)	F 3.0	59-61	N
Prime ALC (See Figure 3-2-10)	ALC SALC (J)	A1	62	A
PQ Order Number	PQ SPQ(J)	A4	63-66	A/N

IC1 = 1 IC1 = 2	Single-indentured LRU Multi-indentured LRU
IC1 = 3 IC1 = 4	SRU Module of an SRU
IC1 = 7	Unit Removed/Replaced at depot only
IC2 = 1 IC2 = 4	Basic item for which requirements are computed Insurance item, not computed by MOD-METRIC

Figure 3-2-1

15
25
35
45
75

C2

DELIVERY SCHEDULE DATA FORMAT

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (Same as Basic Data)	IC1	I1	1	N
Data Type Identifier (must be "5")	IC2	I1	2	N
Blank		1X	3	
Work Unit Code	IDLRU IDSRU (J)	A24	4-11	A/N
Part Number			12-27	A/N
Blank		1X	28	
First Year Deliveries (2 columns per quarter)	QSL (1-4) QSS (J, 1-4)	4I2	29-36	N
Blank		1X	37	
Second Year Deliveries (2 columns per quarter)	QSL (5-8) QSS (J, 5-8)	4I2	38-45	N
Blank		1X	46	
Phase Provisioning (1 column per quarter)	LPP (1-8) SPP (J, 1-8)	8I1	47-54	N
Reserved for future use			55-63	

★

Note: Delivery Schedule data are required for program EVALUATE.

*In RIME, columns 29-39 contain the number of initial provisioning assets for this stock number computed by AFLCR 57-27 rules.

AFLC FORM SEP 63 192E GENERAL PURPOSE DATA SHEET (25 LINES - 5 COLUMNS) PREVIOUS EDITIONS OF THIS FORM WILL BE USED.

11

16
26
36
46
76

C3

STOCK NUMBER DATA FORMAT				
Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (Same as Basic Data)	IC1	I1	1	N
Data Type Identifier (must be "6")	IC2	I1	2	N
Blank		1X	3	
Work Unit Code	IDLRU IDSRU (J)	A24	4-11	A/N
Part Number			12-27	A/N
Blank		1X	28	
Stock Number	NSN SNSN (J)	A16	29-44	A/N
Blank		1X	45	
Commonality Code (C-Common, P-Peculiar)	Not Used	A1	46	A
BSTART for an LRU (Used only on cards with IC1=2, IC2=6)	BSTNEW	F4.2	47-50	N

Note: Stock Number data is optional input to the Mod-Metric program ; except when a unique BSTART is required for an LRU.

Figure 3-2-3

AFLC FORM SEP 63 192E GENERAL PURPOSE DATA SHEET (25 LINES - 5 COLUMNS) PREVIOUS EDITIONS OF THIS FORM WILL BE USED.

C4

TITLE/COMMENT DATA FORMAT

91
92

Data Description	Program Variable Name	FORTRAN Format	Column Numbers	Type Character
Data Type Identifier (must be "9")	IC1	I1	1	N
Data Type Identifier (see below)	IC2	I1	2	N
Blank		IX	3	
Title or Comment	HDG	A69	4-72	A/N

IC2=1 Card is not read by program; used to permit comments or column headings in data file.

IC2=2 Card is read and will be printed as a title on output report.

Figure 3-2-4

c5

Computer Program Parameter Format for GETBSO 95

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (must be "9")	IC1	I1	1	N
Data Type Identifier (must be "5")	IC2	I1	2	N
Blank		1X	3	
Output device designator	NOUT= 06	I2	4-5	N
Input data device designator	INN =07	I2	6-7	N
Desired Buy Support Objective	DBSO =	E12.5	8-19	N

NOTE: This card is required for the GETBSO program only.

Figure 3-2-5

EVALUATOR PARAMETER DATA FORMAT

96

C6

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (must be "9")	IC1	I1	1	N
Data Type Identifier (must be "6")	IC2	I1	2	N
Blank		1X	3	
Delivery Quarters	NQTR	I2	4-5	N
Phase Provisioning Quarters	KPQ	I2	6-7	N
Condemnation Factor	CFAC	F6.2	8-13	N

NOTE: This card is required for the EVALUATE program only.

Figure 3-2-6

15

C7

FLYING HOUR PROGRAM DATA FORMAT 97

Data Description	Program Variable Name	FORTRAN Format	Column Numbers	Type Character
Data Type Identifier (must be "9")	IC1	I1	1	N
Data Type Identifier (must be "7")	IC2	I1	2	N
Blank		1X	3	
Number of bases	NBASES	I3	4-6	N
Flying Hours/Month	FH(K)	F6.0	7-12	N
Order and Ship Time	OST(K)	F3.0	13-15	N

Note: Data in columns 7-15 is for base #1. Columns 16-24 are for base #2. Columns 25-33 are for base #3, etc., for as many bases (NBASES) as there are in the system. If there are more than 6 bases, continue on additional "97" cards starting in column 7.

For the 3-echelon model (TREMOR), use the following for columns 4-39:

Data Description	Program Variable Name	FORTRAN Format	Column Numbers	Type Character
No. of intermediate repair bases (No. groups)	NGPS	I3	4-6	N
No. of satellite bases per group	NSAT	I3	7-9	N
Flying hours at repair base per month	FHMOB	F6.0	10-15	N
Flying hours at each satellite base per month	FHSAT	F6.0	16-21	N
OST days from repair base to satellite base	OSTCM	F6.0	22-27	N
OST days from depot to repair base	OSTDC	F6.0	28-33	N
OST days for aircraft at repair base	OSTCC	F6.0	34-39	N

Figure 3-2-7

16

OUTPUT CONTROL DATA FORMAT 98

C8

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (must be "9")	IC1	I1	1	N
Data Type Identifier (must be "8")	IC2	I2	2	N
Blank		1X	3	
Printer Logical Unit Designator	NOUT	I2	4-5	N
Print Quantity Control	IPRNT	I2	6-7	N
COMBINE data output device	IPNCH	I2	8-9	N
BSO - Stock quantities output device	IBSO	I2	10-11	N

NOTE: If any of the above data items are set to zero or left blank, the output file will not be generated except NOUT which defaults to "06" (printer).

Figure 3-2-8

AFLC FORM SEP 63 192E GENERAL PURPOSE DATA SHEET (25 LINES - 5 COLUMNS) PREVIOUS EDITIONS OF THIS FORM WILL BE USED.

17

C9

COMPUTER PROGRAM PARAMETER FORMAT 99

Data Description	Program Variable Name	FORTRAN Format	Column Number	Type Character
Data Type Identifier (must be "9"	IC1	I1	1	N
Data Type Identifier (must be "9")	IC2	I1	2	N
Blank		IX	3	
Number of bisections	NBIS	I3	4-6	N
Beta	BETA	F6.2	7-12	N
Starting budget factor	BSTART	F6.2	13-18	N
Stopping BBO factor	BSTOP	F6	19-24	N
Condemnation factor	CFAC	F6.2	25-30	N
COMPUTATION CODES				
• BOUND CODE	IMINSK or KMINSK *	I6	37-42	N
• UPPER BOUND	BOMINI or BOMINK *	F6.4	43-48	N
• EQUAL BASE FLAG	IEQBAS or KEQBAS *	I6	49-54	N

*NOTE: IMINSK, BOMINI, AND IEQBAS ARE USED FOR INITIAL PROVISIONING CALCULATIONS. FOR REPLENISHMENT, KMINSK, BOMINK, AND KEQBAS ARE USED.

AFLC FORM SEP 63 192E 3-11 GENERAL PURPOSE DATA SHEET (25 LINES - 5 COLUMNS) PREVIOUS EDITIONS OF THIS FORM WILL BE USED.

18

INPUT /OUTPUT RECORD	D1	SYSTEM NUMBER	D1 PAGE 1 OF 2 PAGES

TITLE GETBSO INPUT RECORD # 1: BSO SUMMARY RECORD

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	BLANK	–			1X
	DATA TYPE VARIABLE #1	IC1			I1
	DATA TYPE VARIABLE #2	IC2			I1
	BLANK	–			1X
	BUY SUPPORT OBJECTIVE	BSO			E12.5
	WORK UNIT CODE OF LRU	ID(1,1)			A4
	" " " " "	ID(1,2)			A4
	NUMBER OF ITEMS IN DATA SET	NU			I3
	TOTAL BACKORDERS FOR THE	BMIN			E12.8
	FOLLOWING LEVELS				
	TOTAL COST FOR THE FOLLOWING	CT			F12.0
	SET OF LEVELS				

AFLC FORM MAR 72 484 REPLACES AFLC FORM 609, WHICH WILL BE USED. AFLC–WPAFB–APR 72 92M

INPUT /OUTPUT RECORD	D2	SYSTEM NUMBER	D2 PAGE 2 OF 2 PAGES

TITLE GETBSO INPUT RECORD #2: BSO LEVELS RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS

NO.	DATA ELEMENT NAME	FORTRAN VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	BLANK	—			1X
	DATA TYPE IDENTIFIER #1 *	IC1			I1
	" " " #2 *	IC2			I1
	BLANK	—			1X
	DATA SET IDENTIFICATION #1				A8
	" " " #2				A16
	NSN	NSN			A16
	ALC	ALC			A1
	TOTAL COST OF THIS SET OF LEVELS	CM			F9.0
	TOTAL STOCK (UNITS) FOR LRU	LTOT			I4
	DEPOT STOCK LEVEL	KD			I3
	DEPOT CONDEMNATION LEVEL	KC			I3
	STOCK LEVEL FOR BASE K, K=1,2,...	NLRUB(K)			23I3

*NOTE. SEE AFLCR 57-13, FIG. 3-2-1 FOR DEFINITIONS.

wJD
8/20/79

INPUT/OUTPUT RECORD	E1	SYSTEM NUMBER RIME	E1 PAGE 1 OF 1 PAGES

FILE: ITEM LEVELS RECORD

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

DATA RECORD LENGTH

TAPE RECORD LENGTH/BLOCKING FACTOR

REMARKS
1. GETBSO OUTPUT
INITM1 INPUT
2. SORTED ON FIELDS 2,3,4,5,&6 IN ASCENDING ORDER

NO.	DATA ELEMENT NAME	RIME VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
–	BLANK	–	1		1X
1	LINE NUMBER	–	2-5		I4
	POUND SIGN (#)	–	6		1H#
	MOD-METRIC IC1, IC2 CODES	IC1 IC2	7-8		I2
	BLANK	–	9		1X
2	IDENTIFICATION CODE	IDENT	10-11		I2
3	λ- NUMBER	NLAM	12-13		I2
4	LRU GROUP NUMBER	LRUGRP	14-15		I2
5	QUARTER NUMBER	KQTR	16-17		I2
6	LRU/SRU NUMBER (LRU=1, ETC.)	NUMB	18-19		I2
–	BLANK	–	20-30		11X
7	OVERHAUL STOCK LEVEL	IOVSL	31-33		I3
–	ALC – (NOT USED IN RIME)	–	34		A1
8	TOTAL COST OF DEPOT, BASE, & CONDEM LEVELS	–	35-43		F9.0
9	TOTAL OF DEPOT & BASE LEVELS	–	44-47		I4
10	DEPOT STOCK LEVEL	KD	48-50		I3
11	CONDEMNATIONS LEVEL	KC	51-53		I3
12	BASE LEVELS, FOR BASE K=1,2,...,NBASES	IBSL(K)	54-56		8I3
			ETC		
			75-77		

INPUT /OUTPUT RECORD	F1	SYSTEM NUMBER	F1 PAGE / OF / PAGES

TITLE RIME LRU GROUP RESULTS RECORD

DATA RECORD LENGTH

TAPE RECORD LENGTH/ BLOCKING FACTOR

REMARKS
1. PUNCHED BY ITRSLT IF ITWRT=1

INPUT / OUTPUT	FILE NUMBER	RUN NUMBER	SEQUENCE OF FILE

NO.	DATA ELEMENT NAME		RIME VARIABLE	POSITION OR BLOCK	STRUCTURE LENGTH	STRUCTURE CHARAC
	LRU GROUP NUMBER		NGROUP	1-2		I2
	INVENTORY POLICY NUMBER		IDENT	3-4		I2
	λ- NUMBER		MLAM	5-6		I2
	REPLICATION NUMBER		IREPL	7-8		I2
	QUARTER NUMBER		I	9-10		I2
		ITRSLT MEASURE NO.				
	LRU (DEPOT) BUY-$	1	IORDER(I,3,3)	11-20		I10
	SRU " " "	2	IORDER(I,3,4)	21-30		I10
	LRU DEPOT BACK-ORDER DAYS	3	IBODAT(I,2,3)	31-36		I6
	SRU " " "	4	IBODAT(I,2,4)			I6
	LRU BASE FILLS	5	IFILLT(I,2,1)			I6
	SRU " "	6	IFILLT(I,2,2)			I6
	LRU BASE REQUISITIONS	7	IREQT(I,2,1)			I6
	SRU " "	8	IREQT(I,2,2)			I6
	LRU BASE BACK-ORDER DAYS	9	IBODAT(I,2,1)			I6
	SRU " " "	10	IBODAT(I,2,2)			I6

AFLC FORM MAR 72 484 REPLACES AFLC FORM 609, WHICH WILL BE USED. AFLC-WPAFB-APR 72 92M